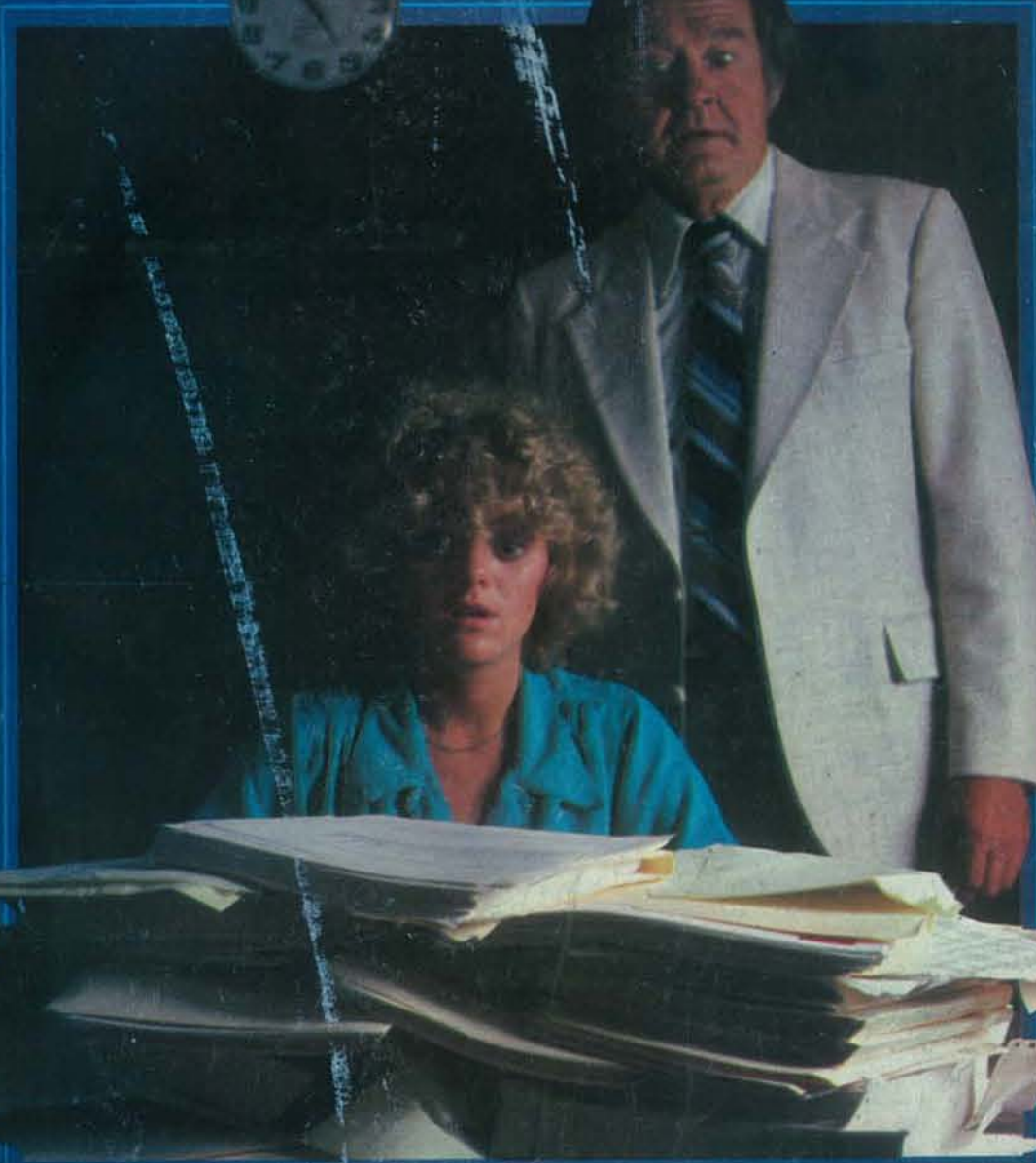


The Evan's Report Is In There – Somewhere!



The old-fashioned mess of paper filing systems can drive you over the edge. You waste time and money in inefficient searching and manual sorting. In today's business environment, this is both outdated and unnecessary.

DATADEX, the information organization system from IUS, hands you your valuable information immediately — when you need it from your APPLE II desktop computer.

You view your stored information on the APPLE™ screen or create reports with simple commands. DATADEX provides just the detailed and professional looking reports needed in today's business world.

With DATADEX, management performance is improved using faster, more accessible information and, best of all, DATADEX has been shown to pay for itself in two months of normal secretarial time. Visit your local dealer and experience DATADEX yourself. IUS, 281 Arlington Ave., Berkeley, CA 94707 (415) 525-9452.

DATADEX The Information Organization System. **IUS**

\$2.50
In Canada \$3.00

Recreational COMPUTING

Adventures in Home Learning and Leisure Vol. 10 No. 2, Issue 53, September-October 1981

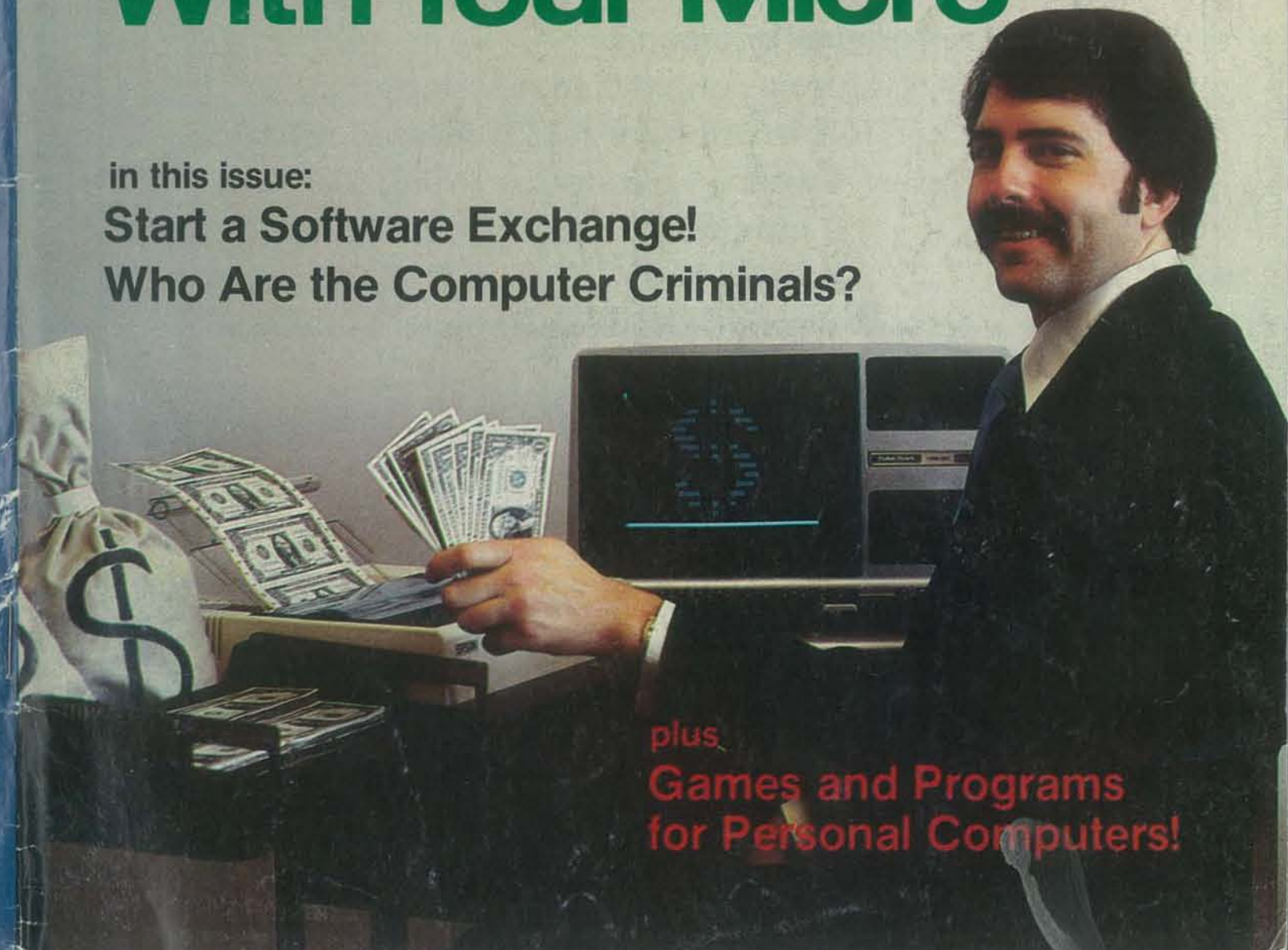
93063JOHNSB07 K2 2276
BYRON B JOHNSON JR
CIVIL ENGINEER JL85
4545-5M INDUSTRIAL ST
SIMI VALLEY, CA 93063

43 Ways to Make Money With Your Micro

in this issue:

Start a Software Exchange!

Who Are the Computer Criminals?



plus
**Games and Programs
for Personal Computers!**

NEW YORK STOCK EXCHANGE: IBM \$89
 AMERICAN STOCK EXCHANGE: HOLIDAY INN \$23
 OVER THE COUNTER : ASCII \$15

SEPTEMBER, 1980

LARRY : YOU GOT YOUR CURRENT SALARY
 OF \$450.00

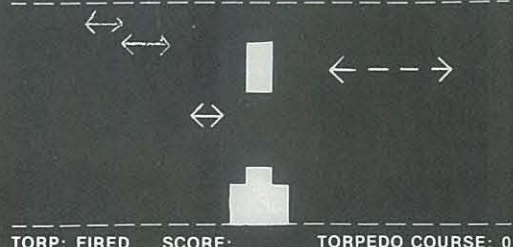
\$40000

RANDI :
 \$57423

BUY: LARRY 400 SHARES OF IBM @ \$89.43

Stock Market

TORPEDOES LEFT: 8 SHIP CONDITION: PEN LEFT: 00



TORP: FIRED SCORE: TORPEDO COURSE: 0

Sea Wolf

NEW! ASCII

for the TRS-80*

Every month you receive a certified ASCII C-20 cassette containing: a cover page with a directory of programs

4 original programs

an information packed newsletter

and information on ASCII funded contests

Rates: 1 year [12 issues] \$40.00

6 months [6 issues] \$25.00

Sample issue \$ 5.00

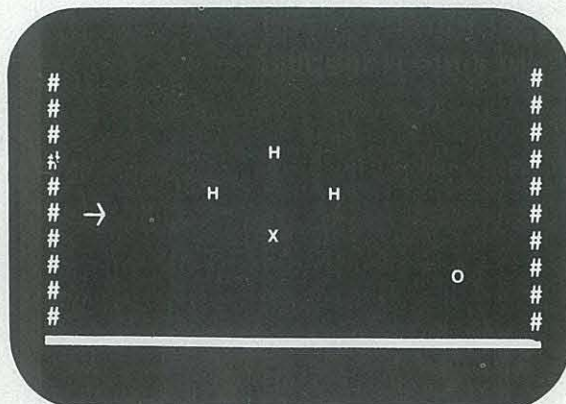
Write For Overseas Rates

To subscribe, write to: ASCII

P.O. Box 516, Valley Stream, N.Y. 11582

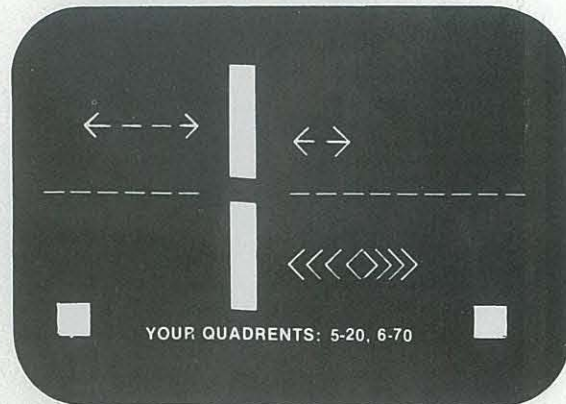
Level II 16K Required

Star Wars Trench



*TRS-80
 is a
 product
 of
 Tandy
 Corp.

Star Fire



Recreational COMPUTING

Editorial

Bringing Up Baby

In our last issue, we provided valuable information about how to select the right personal computer. It's a bewildering feeling to walk into a computer store, knowing next to nothing and confronted with a major purchasing decision. But that's easy compared to the feeling when you open your closet and find a dusty piece of expensive hardware which hasn't been used in six months.

One of the most embarrassing questions you can ask any computer salesman or manufacturer is, "What will I do with it?" He will have a handy reply, of course, but probe his answer. Will it really be easier to balance your checkbook on a computer than with a calculator? If he thinks so, better get a demonstration. Games? Well, they are great, but should you spend a thousand dollars or so to play them? Then there is word processing — if you are an author. It could be a bit extravagant to use a computer to write to mother.

These points are all addressed to the home computer user. Educators have many useful applications for computers, and the software is slowly evolving to fill those areas. Small business software is in roughly the same category: lots of uses, less actual software — but it's coming. Home users are a different story. What will you use your computer for?

Having a home computer is like having your first child. It can do just about anything, but it needs to be taught how to do it. When a baby wants something, it lays in your arms and cries, making you feel helpless until you discover what it needs. When a computer needs something it sits on a table, silently making you feel equally helpless. A child grows up and becomes a responsive member of the family. The computer can become a useful, integral tool in our lives but needs to be expanded. Given some peripherals and some software to suit your application, it will not linger in the attic waiting for the next garage sale.

What is needed is a little imagination. To help spark some creative ideas, we called on Joe Weisbecker to write our cover story for this issue. Joe is the author of *Home Computers Can Make You Rich*. He has followed our magazine for years, and we are pleased to be able to present his work here. You will find he has given many imaginative ideas from which to draw.

Not everyone's imagination works strictly on the up-and-up. Although people mature in their understanding of computers, their ethics do not always keep pace. Computer crime can be blatant or subtle, malicious or mischievous. Paradoxically, small computers can also be used to help rehabilitate prison inmates. Two of our feature articles illuminate these points.

Speaking of legalities, software piracy is a well-worn topic. But did you know that educators have formed a very ethical, legally proper facility called Softswap, which allows access to hundreds of programs? Their example may show a way to get to the root of software piracy and eliminate it by removing its causes. Not even the majority of programs will ever be given away just for copying costs, but having access to public domain software of good quality and reasonable variety could be a very healthy thing for users and manufacturers alike.

We publish software in each of our issues which is for you to use. "Use" is the key word. You can learn a fairly good deal about programming just by typing in the listings we publish and seeing what they do. True to fashion, each computer will have a slightly different version of the BASIC programming language. What will work on one machine will not quite work on another. Some minor tinkering with the lines which cause trouble should get the software to work. How about sharing your modifications?

Our magazine is continually working to make your experience with computers more exciting and less frustrating. Home computers are out of their infancy, but now they are in the formative years. Let us know how the transition is going for you.

Marli Ouwerson

Publisher
 Michael Madaj
Editor
 Marlin Ouwerson
Assistant Editor
 Julie Anton
Contributing Editors
 Dave Caulkins
 Dave Cortesi
 David D. Thornburg
 Ramon Zamora
Art Director
 Clifford West
Artist
 Barbara Ruzgerian

Typesetter
 Renny Wiggins
Proofreader
 Nancy Huebach
Business Manager
 Maureen Christine
Marketing Director
 Craig S. Harper
Advertising Sales
 Kent Safford
Circulation Director
 Peter Clark
Circulation Assistant
 Leah Dansby

Notice to All Foreign Subscribers:

Because the foreign postage rates have increased an average of 18% since January, we've found it necessary to increase the price of foreign subscriptions to cover the new rates. Effective with this issue the new Subscription Rates for *Recreational Computing* are:
 \$22 One Year (6 issues) Canada/
 Mexico First Class Only
 \$30 One Year World Airmail
 \$18 One Year Surface Mail

Subscription rates: \$12.00 per year within the United States; \$22 for first class to Canada and Mexico; \$30 for airmail to other countries. All Foreign Subscriptions must be prepaid in U.S. dollars drawn on a U.S. bank.

Contributing Subscribers: \$25/year (\$13 tax deductible) — Algorithmics, Inc., Gerald Bowman, Robert Connors, Mark Elgin, Joi Ellis, John B. Fried, Scott B. Guthery, William G. Hutchison, Jr., W. A. Kelley, Bill Kling, William M. Richman II, Philip A. Smith, Joseph A. Weisbecker, Brett Wilson, Patrick J. McGovern. **Retaining Subscribers:** \$50/year (\$38 tax deductible) — Dave Caulkins, Robert Connors, Zenith Radio Corp. **Sustaining Subscribers:** \$100+/year (\$88 tax deductible) — Byte Publications; Paul, Lori & Tom Calhoun; David Dick; Louis R. Patzke. **Lifetime Subscriber:** \$900+ (\$700+ tax deductible) — Bill Godbout Electronics. **Corporate Subscriber:** \$500/year (\$440 tax deductible).

Advertising: Advertising space is available in this publication. Please direct inquiries to the Advertising Director, *Recreational Computing*, Box E, Menlo Park, CA 94025, 415/323-3111.

Foreign Distributors of *Recreational Computing*: UK & Europe: L P Enterprises, 8/11 Cambridge House, Cambridge Road, Barking, Essex, IG11 8NT, Great Britain. Hofacker-Verlag, Tegernseer Strasse 18, D-8150 Holzkirchen, West Germany. **Asia & Australia:** Computer Store, POB 31-261, 22B Milford Road, Milford, Auckland 9, New Zealand. ASCII Publishing, 305 Hi Torio, 5-6-7 Minami Aoyama, Minato-ku, Tokyo 107, Japan. 261, 22B Milford Road, Milford, Auckland 9, New Zealand. ASCII Publishing, 305 Hi Torio, 5-6-7 Minami Aoyama, Minato-ku, Tokyo 107, Japan.

Atari is hot and so are we

EVERYONE IS TALKING ABOUT THE ATARI® 800™ COMPUTER. Color graphics and English characters with truly high resolution; high quality sound; expandable memory; and sleek, modular appearance have made the 800 a "must have" for many computer users. And when you add ATARI's full line of advanced peripheral components and comprehensive software library you get a really hot deal, whether your application is business, professional or personal.

SPECIAL OFFER — ASAP makes the ATARI® 800™ the hottest deal in town by offering 16K bytes of additional RAM . . . FREE! You get 32K for the price of 16K at ASAP's special price of only \$799.00. So don't get caught out in the cold. Call ASAP today.

COMPLETE SOFTWARE LIBRARY INCLUDES THESE POPULAR UNITS:

Basketball ROM	\$ 30.00
Super Breakout ROM	\$ 30.00
Educational System ROM	\$ 19.95
Video Easel ROM	\$ 30.00
Music Composer ROM	\$ 45.00
Computer Chess ROM	\$ 30.00
3D Tic Tac Toe ROM	\$ 30.00
Star Raiders ROM	\$ 34.00
Kingdom Cassette	\$ 12.95
Blackjack Cassette	\$ 12.95
Biorhythm Cassette	\$ 12.95
Assembler/Editor ROM	\$ 45.00
Mortgage Loan Analysis Tape	\$ 12.95
Stock Analysis Disk	\$ 19.95
Charting Disk	\$ 19.95
Bond Analysis Disk	\$ 19.95
Mailing List Tape	\$ 17.95
Touch Typing (2 tapes)	\$ 19.95
Graph It (2 tapes)	\$ 17.95
Energy Czar Tape	\$ 12.95
Telelink (Terminal ROM)	\$ 19.95
Space Invaders Cassette	\$ 15.95
*Scram	\$ 16.95
*Word Processor	\$ 99.95
*Asteroids	Call for price
*Missile Command	Call for price

*New item



OPTIONAL ACCESSORIES

ATARI® 410™ Program Recorder	\$ 60.00
ATARI® 810™ Disk Drive	\$ 475.00
ATARI® 815™ Dual Disk Drive	\$1195.00
ATARI® 820™ 40-column Dot Matrix Impact Printer	\$ 279.00
ATARI® 822™ Thermal Printer	\$ 349.00
ATARI® 825™ 80-column Dot Matrix Impact Printer	\$ 625.00
ATARI® 830™ Acoustic Modem	\$ 159.00
ATARI® 850™ Interface Module	\$ 139.00
ATARI® Paddle (CX30-04) and Joystick (CX40-04)	\$ 17.95
Light Pen (CX-70)	\$ 64.95

ASAP offers a 120-day buyer protection policy: full money-back guarantee if not totally satisfied.

Ordering Information: name, address, phone, ship by: UPS or Mail. Shipping charge: add \$2.50 up to 1 lb. (UPS blue), U.S. Mail add \$1.50 (U.S. only) (\$25.00 minimum order).

Terms: We accept cash, check, money orders, Visa and Master Charge (U.S. funds only). Tax: 6% Calif. Res., COD's and terms available on approval (School PO's Accepted).

Toll free outside California: (800) 421-7701 Inside California: (213) 595-6431 (714) 891-2663

asap
computer
products, inc.

1198 E. Willow St., Signal Hill, CA 90806

Recreational COMPUTING

September/October 1981, Volume 10, Number 2, Issue 53

Learning About Computers

- 8 43 Ways to Make Money with your Micro** by Joe Weisbecker
You say you can't justify spending that much money to play games or to balance your checkbook? Feeling guilty about that unused computer in the closet? Here are lots of practical ways to put your computer to work earning hard cash!
- 16 Who Are the Computer Criminals?** by Jay BloomBecker
They don't wear dark masks over their heads or gloves to hide their fingerprints. These criminals have more refined techniques than a cat burglar. With both large and small computers they can transfer funds to their checking accounts, acquire gems, and get secret information. How do they do it — and why?
- 20 Micros Behind Bars** by Mike Briggs
On the other hand, computers are what you make them. This inmate tells how he uses his computer, and the hope it has given him for his eventual return to society.
- 24 How to Start a Software Exchange** by Ann Lathrop and Bobby Goodson
The cost of software got you down? Worried about the piracy issue? Want to try before you buy? A group of dedicated educators has formed a modern trading post of good quality, reliable software. You can use their resources — or you could follow their example and start a local software exchange!
- 44 Number Systems** by Alfred J. Bruey
If binary, hexadecimal and octal sound like forms of marine wildlife to you, then this article is definitely useful. This simple yet complete explanation shows what counts in the world of computer programming.



8



16



44

Games and Programs

- 30 Computer Knock-Knock Jokes** by David J. Beard
The author provides programs in BASIC and in LISP for those who like corny jokes, or who are intrigued by conversational computer programs.
- 32 For Photographers Only!** by Joey Robichaux
If you enjoy photography, read this article. You can learn to use your personal computer to generate helpful charts and tables which can make your hobby more fun and could improve your photographs as well!
- 34 Fibonacci Nim** by Mary Ann Gohlke
This version of the classic Nim game is perfect for days when your computer supports a steady stream of curious people. The program is friendly and will play with several players at once — but how many of them will win?
- 38 Roman Numeral Conversum Program** by Ralph Roberts
This author *knows* why the Roman Empire crashed — his program will help prevent it happening again. Teach your computer to convert those alphabets to numerics.

Departments

- | | |
|--------------------------|-------------------------------------|
| 3 Editorial | 27 Programming Problems & Solutions |
| 6 ComputerTown, USA! | 30, 32, 34, 38 Games & Programs |
| 14 Classroom Connections | 40 Pocket Corner 42 Product News |

Cover by PCC Productions

RECREATIONAL COMPUTING (ISSN #0164-5846) is published bimonthly by People's Computer Company, 1263 El Camino Real, Box E, Menlo Park, CA 94025. People's Computer Company is a non-profit, educational corporation. Donations are tax deductible. Second class postage paid at Menlo Park, California, and additional entry points. Address correction requested. Postmaster: send form 3579 to Box E, Menlo Park, CA 94025. Copyright 1981 by People's Computer Company, Menlo Park, California.

Securing Your PET

ComputerTown, USA! recently had the misfortune of being victimized by a sneaky, small-time vandal at our facility in the Menlo Park Public Library. There was no real damage to any of the machines. In fact, the only problem we have had so far is that someone keeps taking the fuses and their caps from the back of the PETs. For awhile, we were buying replacement caps and fuses and, as soon as we got them installed, someone would come along and remove them again. Now, I don't have the faintest idea what one would do with all those caps and fuses, but I do know that the computers don't work at all without them. So, we have come up with a minor modification for the "old-style" PET that moves the fuse and its holder inside the machine. Now our tormentor will have to take the entire top off the computer to get at the fuse. I don't think they will go this far since the machines are in clear view of the library desk and the act of removing the cover is sure to attract attention.

The modification is a simple one that you can do with a few tools. If you don't want to do it yourself, I'm sure that your PET dealer, a local PET user or the nearest TV repair shop will be happy to help. The tools you will need are: a Phillips screwdriver, a nut driver (1/4" for most machines), a diagonal wire cutter and a wire stripper. Your supplies are: two solder-less "wire nuts" and a new in-line fuse holder. The fuse holder must have its own wire leads that can be spliced to the wires that are now connected to the existing fuse holder. There are some in-line holders that are supposed to be "quick connecting" to the existing wires, needing no wires of their own. Unfortunately, the wires that are connected to the computer's fuse holder are too short to accommodate this type of holder. So, get the one with its own wires attached.

Unplug your computer before you begin this procedure!

Begin by using the Phillips screwdriver to remove the screws that hold down the front of the computer's cover. They are located on the sides of the machine, near the front and just below the lip of the upper housing.

Once the screws are removed, open the case and tilt the upper housing as far back as you can. You should unplug the

keyboard wire ribbon, the video monitor and the tape drive to allow the housing to tilt back even further. Carefully observe the way these ribbon wires are inserted in their respective receptacles so that you can replace them the same way.

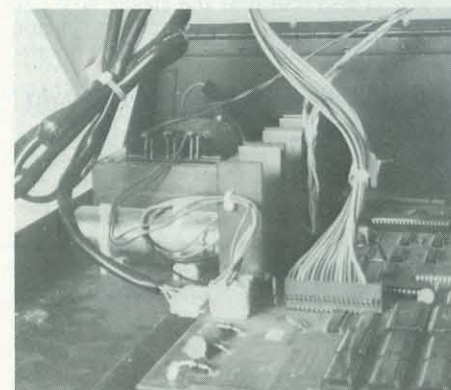
As you face the open machine from the keyboard position, locate a large transformer near the left-rear corner of the lower section of the case. It is a large,



The PET computer . . .



. . . its original fuse location . . .



. . . and the new location.

heavy-looking block that appears to be partially wrapped in paper and has several wires attached to it. Immediately behind the transformer you will find a black metal plate. It is attached to the left side wall and bolted to the bottom of the lower housing. Use the nut driver to remove the nut from the bolt that holds this plate down.

Having removed the plate, locate the backside of the computer's ON-OFF switch, of the fuseholder and of the power cord. Notice that the fuseholder is wired to both the power cord and the switch. One wire leads from each of them to soldered connections on the fuseholder. Use the diagonal cutter to carefully snip the connecting wires as close to the fuseholder's connections as possible.

Use the wirestripper to remove about 1/4 inch of insulation from the newly cut wires.

Take the new fuseholder in hand and use the diagonal cutter to snip each of the wire leads to about 3 inches long. Use the wire stripper to remove about 1/4 inch of insulation from each of the wire leads.

Place one lead from the new fuseholder beside one of the computer's wires so that the ends of the wires are even. Insert the two wire ends into one of the wire nuts and turn the nut clockwise. This has the effect of "screwing" the nut onto the wires. The wires are spliced and insulated in this way. Now take the other fuseholder lead and computer wire pair and splice them with the other wire nut.

That completes the installation. There is no real need to replace the metal plate, but you can if you want to. Be sure to reconnect the video monitor, power supply and keyboard to their proper place before closing the cover and replacing the screws that hold it down.

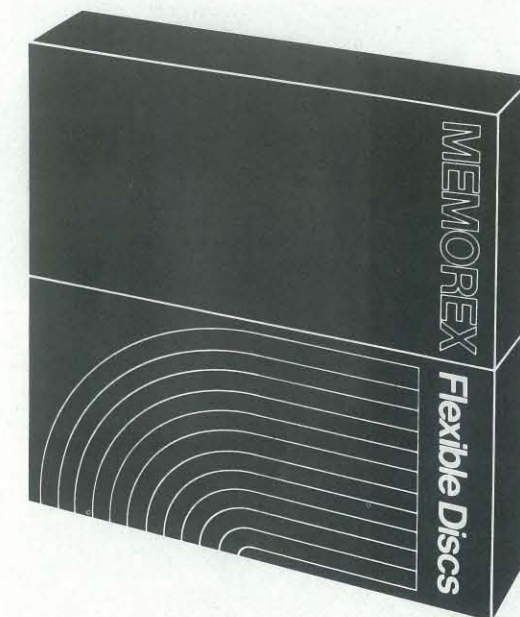
Please note: this modification will undoubtedly void your computer's warranty. You will have to weigh this eventuality against the need for fuse security.

Reprinted by permission from the ComputerTown, USA! News Bulletin. ComputerTown, USA! - A computer literacy project of the people of Menlo Park, CA, the Menlo Park Public Library, and People's Computer Company, a non-profit educational corporation. P.O. Box E, Menlo Park, CA 94025. Donations are tax-deductible.

"This material is based upon work supported by the National Science Foundation under Grant No. SED 8015964. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation."

MEMOREX

SUPERB QUALITY AT UNBEATABLE PRICES. THAT'S MEMOREX DISCS FROM PACIFIC EXCHANGES



For Data Reliability—
Memorex
Flexible
Discs

CALL FREE FOR PRICES & INFORMATION

(800)235-4137

In California, (800)529-5935 or (805)543-1037

DEALER INQUIRIES INVITED



C.O.D.'s ACCEPTED



PACIFIC EXCHANGES
100 Foothill Blvd.
San Luis Obispo, CA
93401



WAYS TO MAKE MONEY With Your Micro



Joe Weisbecker

The microcomputer field provides hundreds of new ways to make extra money at home in your spare time. Unfortunately, most articles and books emphasize business applications of microcomputers. While many experienced programmers are making money as small business consultants, this approach isn't practical for the average home computer owner. To successfully pursue small business applications you need an expensive system, considerable experience, and more free time than most of us have available. You must also worry about deadlines and unhappy customers. There are probably enough problems in your regular job without asking for more in your spare time.

Fortunately, there are many ways to make money with microcomputers that don't involve conventional business applications. With a little imagination your computer hobby can be profitable as well as fun. I never buy a computer without deciding in advance how it can be used to pay for itself. This should also be a primary consideration in your choice of a personal computer. By letting your computer pay for itself, you can enjoy it without feeling guilty about the expense. You can even afford to own several computers!

GETTING STARTED

Your first step toward spare time microcomputer income involves starting a list of money-making ideas. You should always be looking for these ideas. You can find them by reading books and magazines or by talking to friends. Thinking about products or services that you would like to have available will give you more ideas.

Sooner or later you'll find something that's just right for you. Never pursue an approach that you don't feel comfortable with. Money-making ideas that are related to your existing interests or hobbies are the best kind. You'll have an edge over your competition. An interest in fantasy gaming or bridge can be converted into salable computer programs. A teacher might be able to produce exceptional educational programs in her spare time.

This article provides a few potentially profitable ideas for you to start thinking about. Many of these only require a minimal amount of hardware. Some require none at all. Other ideas can be pursued by beginners since skill is secondary to imagination. My book, *Home Computers Can Make You Rich* (published by Hayden Book Company) describes many other spare time money-making approaches. Ideas involving articles, books, newsletters, programs, games gadgets, tools, arts, crafts, attachments, computer-made products, inventing, investing, and services are discussed.

Copyright © 1981 by Joe Weisbecker.
All rights reserved.

There are really only four basic ways to make money in any field. You can sell products, sell services, create new products, or invest. Some ideas in each category will be described. Various combinations of these four approaches are possible. But before getting down to specific ideas, let's consider how you will turn your idea into income.

SELLING PRODUCTS

Mail order selling involves placing advertisements for your product in various publications and filling the resulting orders by mail. A small mail order business can easily be operated out of your home. This is an attractive spare time business if you have the right product and a good ad. As a first step you should read several books on the mail order business in general. These can be found in almost any library or bookstore. It is possible to start a business of this type with a minimum investment. You'll need a post office box, a product source, and enough money for initial advertising. Advertising will be your biggest expense. A single, small advertisement can cost several hundred dollars.

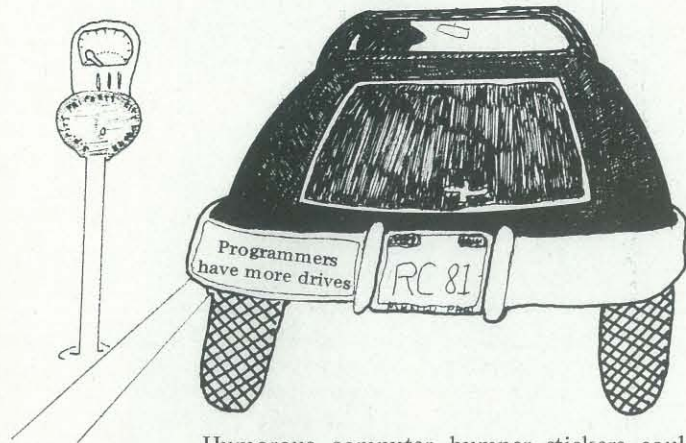
A low-cost, easy-to-mail product is ideal. It should not be available locally in stores. If it is unusual, you might be able to get it mentioned for free in some computer magazines.

The price you charge must include the cost of advertising. Suppose that your product is a booklet that costs \$1 to print and fifty cents to mail. Let's assume that your advertisement in a magazine costs \$250 and the magazine has a circulation of 100,000. You might estimate receiving 100 orders from this advertisement. Your selling price for the booklet would have to be \$4 just to break even. This doesn't even include the cost of your time. If your ad pulled less than 100 orders you would lose money. If you got 200 orders you would make \$250 (not including your time). The effectiveness of your advertisement can make the difference between success and failure.

The secret to a successful mail order business involves picking the right product and testing your advertising. You should run several different ads to see which one generates the most orders. You can specify Box 100-A as the return address in one ad and Box 100-B in another to measure the returns from each. Place small ads first so you can see if there is any interest in your product before spending a lot of money. You might also want to test various prices for your product. A higher price sometimes turns out to be more profitable.

In order to pick a product, to slant your advertising properly and to choose the best places to advertise, you must understand your potential customers. Hardware hackers, programmers, and unsophisticated users are the three general types of

customers for your product or service in the microcomputer field. By examining the various computer publications you'll easily discover which types of customers they're aimed at. If you see lots of circuit diagrams then it's a publication aimed at hackers. Articles discussing operating systems and programming techniques appeal to programmers. General articles and ready-to-run program listings are slanted toward users. Some products such as T-shirts, bumper stickers, and computer jewelry might appeal to all three types of customers.



Humorous computer bumper stickers could provide a nice mail order product. They are inexpensive to make and mail. You don't even need a computer. All you need is a sense of humor and some imagination. If you are an engineer or programmer then you'll need to find a friend with a sense of humor to think up ideas for you. This market should grow as the interest in computers increases.

Unique, computer-related greeting cards could be designed and sold by mail. Computer owners should be good customers for this type of item. Computer-related posters and jig saw puzzles could also be sold by mail. No computer expertise is required to sell this type of product.

Selling information by mail is a popular business. Information on consulting and lists of program publishers are currently being successfully sold by mail. Booklets of programs for small computers such as the Sharp pocket computer or the Sinclair ZX80 are nice mail order items.

Local instant printing shops can print booklets for you. They are fast and inexpensive. You can prepare the information yourself or pay someone to write it for you. You might provide a guide to stock market or gambling systems. A booklet of ideas for computer science experiments might sell. Information for specific applications, attachments, or hardware modifications can be sold by mail. The possibilities are endless. You'll never know if you can come up with a winner unless you try!

Other mail order opportunities exist if you own a popular type of computer. You could design and sell customized, animated video graph-

ics for birthdays, anniversaries, etc. Color and sound would be an added attraction. Your market would be unsophisticated owners of consumer-type computers. They would receive a cassette containing the ready-to-run customized program.

Along the same lines, you might provide a questionnaire. Parents would fill in their child's name, friends, pets, and interests. This information would be used in a personalized computer story or adventure game. The parents would buy the resulting cassette. Again, you should only attempt this with a very popular computer aimed at consumers in order to maximize your potential market.

By now you should be starting to think of other products you could sell by mail. Add them to your list so you won't forget them. Ideas that don't seem practical now might be useful in the future as microcomputers become cheaper and the home computer market expands. In particular try to think of new ideas for personalized products.

SELLING SERVICES

You can provide a wide variety of local computer services in your spare time. None of the services discussed here involve business consulting. Some of these services require little or no skill while others require some programming ability.

Renting your computer to people is one type of local service you could provide. Look for applications where someone only wants temporary use of a computer. If they want a computer for long-term use they will probably buy one instead of renting it. People will really be paying you a small fee for short-term use of a specific program.

The largest number of programs are available for the original Radio Shack computer (Model 1, Level 2 BASIC). Since this machine is no longer in production you might eventually be able to obtain second hand units at bargain prices. This would be an ideal way to start a small home computer rental business with a minimal hardware investment.

Following are some rental service ideas. You should be able to think of many others. Such services can be advertised in the classified section of your local newspaper. Be sure to specify what the computer will do for people.

Children's birthday parties always need novel forms of entertainment. Your computer could provide a variety of video games for a reasonable price. Parents might also be interested in renting a computer to keep an ill child occupied. It is important to provide a suitable selection of game programs along with the computer. There are even adult game programs available if you don't want to limit yourself to children's parties.

Elderly patients in rest homes are often lonely and would welcome new interests. They might be



interested in renting a computer for its entertainment value. Adventure games should be ideal in this environment.

Skill sharpening provides another reason for someone to rent your computer. Special drill and test programs can be valuable for tutoring children who need extra help in a specific area. Someone might want to rent a computer just long enough to learn BASIC so that they can apply for a better job. There are programs that can be used to learn touch typing, bridge, and morse code. Blackjack programs can be used for card-counting practice. Software to help students prepare for SAT and PSAT exams is available. All of these applications provide opportunities to rent your computer.

You could also rent computers to people who are interested in buying one but want to see if they like it first. This is the same approach used by people who rent pianos to see if their children like taking lessons before making a major investment.

Other types of services would depend on your skills. Teaching various computer languages might be profitable. This could be done on your own or for an existing evening adult education program in your neighborhood.

If you can program, it is possible to design customized, animated video displays for local stores to use during special sales, etc. You would have to program some examples and determine if



there is any interest. Stores in shopping malls should be interested in new, eye-catching, window displays. Color would be an added advantage for this application.

Another type of service involves demonstrating computers in people's homes. It should be possible for you to make an arrangement with a local computer store. You would earn a commission on any computer that is sold as a result of your home demonstration. You could even arrange home computer parties similar to those used to sell kitchenware or cosmetics. An entertaining presentation and interesting software will increase your chances for a sale. You could also offer potential customers a rental unit. Seeing how much fun they can have will help them decide to buy their own personal computer.

WRITING PROGRAMS FOR PROFIT

Writing a program is one way to create a new product. You can sell this new product yourself or have someone else sell it for you.

First you must decide which computers to create new programs for. This choice doesn't have anything to do directly with the features or specifications of various computers. You will generally be most successful writing programs for the smallest version of the most popular computers. It's easy to see why. Suppose one type of computer is owned by 200,000 people and another brand is only owned by 20,000 people. You will be lucky if one percent of the owners buy your program. Would you rather sell 2,000 copies of your program or 200? Look at software advertisements in magazines to see which computers are most popular with program publishers.

The second decision you must make is how to market your program. You could sell the program yourself by mail. This involves making copies on cassettes or disk, preparing documentation, advertising, filling orders, etc. You will probably be better off letting an existing publisher manufacture and sell the program for you. This will give you free time in which to develop other programs. The publisher also assumes most of the risk for you. All you can lose is some time. You can find program publishers by looking at software ads in magazines. This will also give you some idea of how much advertising each publisher does. Larger publishers are less likely to go out of business while smaller ones might be more willing to buy your program.

You can sell your program to a publisher either on a cash or royalty basis. You are usually better off with a royalty arrangement. Royalty rates vary from publisher to publisher. You can expect to receive at least ten percent of the publishers' income from your program. Royalties are



generally paid on a quarterly basis but some publishers pay monthly royalties.

Some general guidelines should be followed for any type of program you develop. All of the program publishers that I've talked to like graphics. You should try to use as much animation as possible in your programs to enhance their sales appeal. Programs like Android Nim and Dancing Demon owe most of their success to clever graphics.

You should also concentrate on programs that don't require lengthy explanations. Programs similar to popular arcade games sell well because potential customers immediately recognize what they do. Without this instant recognition factor your programs will be much harder to sell.

Here are a few ideas for non-business programs that you might work on. There is always a market for new game and novelty software. Adventure games have become very popular and you could try to think of some imaginative new scenarios. Can you incorporate more of an educational aspect? Humor, color, sound and speech can be used to enhance all types of games.

Animated TV puppets for young children to play with should appeal to parents. Color and sound make this type of electronic busy box even more attractive. This could be a good way to develop eye-hand coordination.

You might want to try your hand at developing new types of programming languages. Easy-to-use languages for animated graphics or sound effects should be salable.

All sorts of educational programs are possible. There is a major shortage of imaginative programs

in this area. Needs range from pre-school to adult education. Talking computers should be ideal for teaching young children how to read. Adult education software could be aimed at teaching money management or programming. Simulation programs of all types fall into the category of educational software.

Better stock market investment software is needed. New stock or option trading systems that provide automatic buy and sell signals could be developed. These and other programs should be designed for use with on-line data bases.

Health applications is a wide open field. Can a computer program reduce stress by using hypnosis techniques? Perhaps you can develop software for customized exercise or for calculating needed vitamin supplements based on family eating habits.

SPARE TIME WRITING

Computer magazines always need new material that will interest their readers. Some of these magazines pay up to \$50 per printed page. Writing for these magazines can help pay for your computer hobby. Even writing for magazines that don't pay can provide you with valuable experience and prestige. Be sure to read several recent issues of any magazine that you intend writing articles for. This will give you a good idea of the style and type of article the magazine will buy. Each magazine is different and something written for one magazine won't necessarily appeal to another. Slanting your writing toward a specific publication will increase your chance of success.

Most magazines have a free writer's guide available if you ask. Article submissions should be typed double-spaced with your name on each page. When you submit an article to a magazine be sure to include a self-addressed return envelope with the proper postage. Most magazines buy all rights to your article. Read a book on copyright law before signing any contract. You can write a brief letter asking if there is any interest in the type of article you're writing before you submit it. Never submit the same article to two publishers at the same time.

There are many types of articles that you might consider writing. Novel microcomputer applications make good article material. You can describe what you are doing with your computer or what others are doing. Short programs or programming techniques that you have developed can form the basis for profitable articles.

Some magazines are interested in fiction or profiles of interesting people in the computer field. Tutorial articles on various aspects of personal computing are salable if well done. You should try to develop new, imaginative teaching approaches for this kind of article.

Hardware articles can describe computer modifications or homebrew attachments. Not all magazines will buy hardware articles but those that do often pay well. Critiques of new hardware, languages, or programs can be sold if they are timely and objective. You should remember that most magazines make a profit from advertisements. As a practical matter it is better to write critiques of products you like.

You should subscribe to as many computer magazines as you can afford. They will keep you aware of current trends in the microcomputer field and provide lots of ideas for articles. In addition to articles, many computer magazines pay for book reviews. If you have any artistic talent and a sense of humor you might be able to sell computer cartoons to various publications.

You don't have to limit yourself to writing for computer magazines. You could also put your computer knowledge to work by writing for general interest publications. Men's magazines might be interested in how microcomputers are being used in automobiles, robots, and home appliances. Women's magazines might be interested in new computer career opportunities. The subject of microcomputers as a hobby could provide articles for magazines aimed at retirees or parents. The growing use of microcomputers in education provides even more material for general interest articles.

INVESTING

Your personal computer can be used in managing your investments. Software is available to analyze stock prices and help you to decide when to buy and sell. You can also obtain programs that let you evaluate a wide range of put and call option strategies.

Instead of using your computer to analyze investments, you can invest in microcomputers. Stocks of microcomputer or software companies can provide excellent investment opportunities. The price of Cut & Curl, Inc. went from \$5 per share to over \$15 per share this year when they announced that they were planning on opening a chain of "software supermarkets."

Earlier this year Tandy Corporation (Radio Shack) stock was selling as high as \$30 a share. In the last half of 1977, when Radio Shack announced the TRS-80 computer, you could have bought this stock at an equivalent price of \$4 per share. If you wanted more than a 600% gain in four years you could have gotten it by buying and selling Tandy call options.

With computers as your hobby you are in a good position to uncover microcomputer-related investment opportunities. Look for companies with products or services that fill a real need. The robot field should also offer some promising new

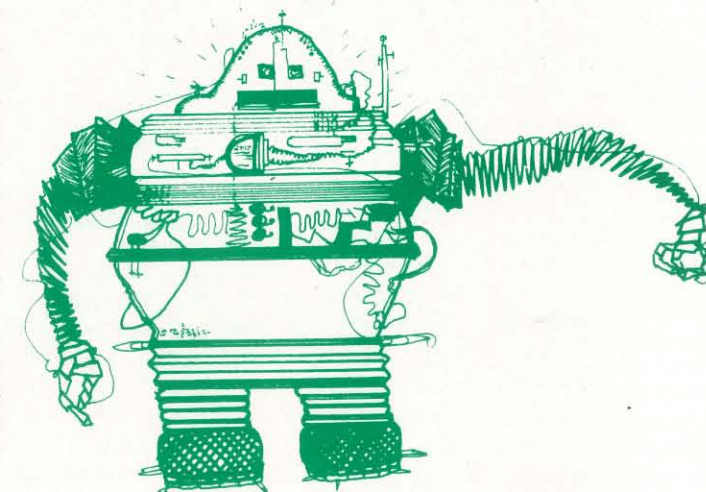
investment opportunities during the 80's.

A third way to invest money is in yourself. Investing in books, courses, and magazine subscriptions are ways of increasing your money-making skills. Suppose that you get an idea for a \$50 article from a \$5 book. This is a 900% return on your investment. It's hard to think of any better ways to invest small amounts of money.

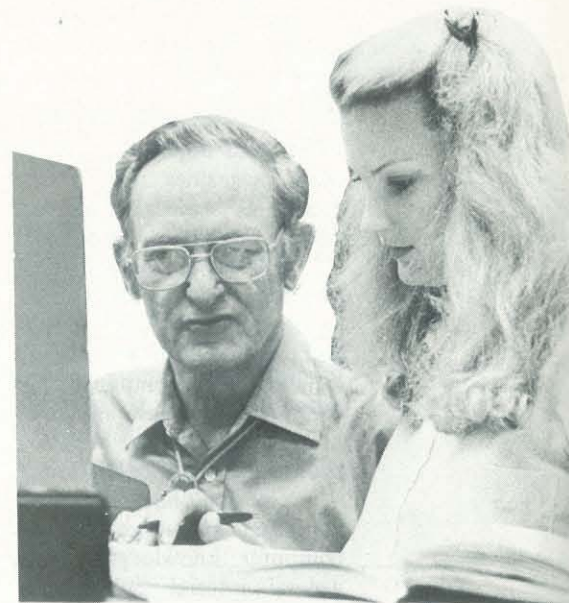
CONCLUSION

By now you should realize that microcomputers really do provide a wide variety of spare time money-making opportunities. We have discussed ten mail order product ideas and over a dozen services that you might provide. Eight program ideas, ten related to writing, and three investment approaches brings the total to forty-three. It's up to you to expand this list and put the ideas that appeal to you to work. You will find out that making money can be even more fun than spending it. ■

Joe Weisbecker is a successful freelance writer and inventor as well as a professional computer scientist. As a personal computer pioneer, he has been designing and using computers for 25 years. He holds 25 patents, has had 2 books published, and is a senior member of the IEEE. 18 of his articles have been published in magazines such as Byte, Creative Computing, Popular Electronics, and Magick. Awards include IEEE COMPUTER best paper award and RCA David Sarnoff award for outstanding technical achievement. Joe has created a wide range of new products including a popular 8-bit microprocessor, several microprocessor support chips, one of the first programmable video games, several commercial hobby computers, a new interpretive programming language, various toys and games, lots of software, and even a greeting card. Hobbies include the stock market, meditation, beard growing, mentalism.



Computer Workshops for Mathematically Gifted Children



Dana Hankins, Marietta, Oklahoma High School student seeks help from Dr. Richard V. Andree, professor of mathematics and of computing sciences at a recent two-week *Computing and Problem Solving Workshop for Mathematically Gifted Students* held at the University of Oklahoma each Spring. Sessions last from 9 a.m. until 9:30 p.m. with both group and individual instruction available for all.

Imagine six dozen mathematically gifted teenagers gathered for two weeks of intensive instruction on computer-assisted problem solving! "It was fascinating," said Dr. Richard V. Andree, professor of mathematics and computing sciences at the University of Oklahoma. Andree and his wife, Josephine Peet Andree, who are well-known for their outstanding work with gifted, creative and talented youth, conducted the second annual *Free Computing and Problem Solving Workshop for Mathematically Gifted Students* at the University of Oklahoma, Norman. Two weeks were devoted to intensive work on microcomputer programming and computer-assisted problem solving. Participants were selected from among gifted students who will be returning to their school system next year and who had completed a year of mathematics beyond first year algebra. Students had to be sufficiently gifted so their principals permitted them to miss school to attend this outstanding program, because May 25-June 5 was the only time the extensive microcomputing facilities of the University could be made available exclusively for this group. Mini (VAX and PDP1170) and Macro (IBM 360/158) computers were also available for individual use, but most of the work concentrated on TRS-80 microcomputers which are easily available to school systems. Students who were accepted could invite a parent or teacher to accompany them. Schools were limited to 1 or 2 participants unless they could furnish extra computers for additional applicants. Originally planned for 50 participants, additional computers were rounded up by participants so 73 persons could finally be accepted. Only students with outstanding records for reliability, deportment and honesty as well as high mathematical ability were considered.

Participants attended 4 hours of hands-on computer lecture-demonstration and from 4 to 8 hours of supervised laboratory during each day of the intensive 9 a.m. - 9:30 p.m., ten-day program. Dr. Andree and Mrs. Andree, mathematical educators, were assisted by six laboratory assistants selected from advanced gifted students at the University of Oklahoma.

The workshops, part of an extensive program for both the secondary and particularly the college level, is conducted each Spring at the University of Oklahoma with financial assistance from the University of Oklahoma Associates.

The outstanding problem-solving ability displayed by the group enabled them to produce numerous solutions to a set of seventy-five difficult micro-research problems proposed in the text *Explore Computing with TRS-80 and Common Sense* (Prentice-Hall, 1981). A few sample problems include:

1. Determine all $N < 1,000,000,000$ such that $N*N$ ends in N . Example: $N = 76$, since $76*76 = 5776$. [Some programs required overnight computer time, others generated all such N in less than two minutes run time.]
16. Find a "Pythagorean box" having integers (whole numbers) for length, width, height and all six surface diagonals. [Many such boxes were found. None were found that also had integral internal (body) diagonals. No one knows if they exist or not.]
20. Write a program to create Haiku poetry. [Solutions included 3 programs to write English Haikus, and one to create Haiku poetry in Japanese.]
39. Compute and print the number $2 \uparrow 75$ exactly, correct to the last digit.

[Several different methods were developed - each of which produced the correct result.]

49. Design a program to coach emergency first-aid treatment at the Boy Scout level. The program should ask questions and then prescribe action based on the response. [This was fun. Several unexpected results occurred to our highly imaginative young programmers.]

Other problems successfully solved by members of this gifted group include problems on amicable pairs, sociable chains, monkeys on typewriters, cryptography, and the eight queens problem. Each problem in the 75 involves special difficulties for beginners on a microcomputer.

Although the majority of participants were from the state of Oklahoma, students from three other states were also involved. The Andrees hope next year's applicants will come from even further distances. Low-cost dormitory housing will be available for non-commuting participants.

Many of this year's participants plan to have programs to display at the day-long *Computer Science Student Fair and Brag Session* which will be held on Saturday, October 17, 1981, in the Physical Science Center at the University of Oklahoma, Norman. The *Fair and Brag Session* is open to student computing buffs of any age and from any location. Last year's fair brought students from 6th grade to junior college level. The only prerequisite is a knowledge of either programming or electronics and an interest in sharing knowledge with other computer buffs. Write or phone Dr. Andree, Mathematics Department, The University of Oklahoma, Norman, OK 73019 - (405) 325-3410 for an application and further details about either program. ■

COMPETENCY EXAM PREPARATION SERIES

This comprehensive set of programs consists of simulated exam modules, a thorough diagnostic package, and a complete set of instructional programs. It is designed to teach concepts and operations, provide drill and practice and assess achievement levels through pre and post testing. The Competency Exam Preparation Series provides a structured, sequential, curriculum encompassing mathematical, reading and writing instruction.

The C.E.P.S. program is designed for individual student use or use in a classroom setting. Programs provide optional printer capability, worksheet generation and performance monitoring. C.E.P.S. are available in two software formats.

National Proficiency Series	\$1,299.00
N.Y.S. Regents Competency Test, Preparation Series	\$1,299.00

If desired separate Mathematics and Verbal packages are available for \$799.00 ea. A Spanish language version of the Mathematics Instruction Package is available at no extra charge.

COLLEGE BOARD PREPARATION SERIES 81/82 for TRS-80 NORTHSTAR™ PET, APPLE OSI

Each program confronts the user with a virtually limitless series of questions and answers. Each is based on past exams and presents material of the same level of difficulty and in the same form used in the S.A.T. Scoring is provided in accordance with the formula used by College Boards.

S.A.T., P.S.A.T., N.M.S.Q.T., set includes 25 programs covering Vocabulary, Word Relationships, Reading Comprehension, Sentence Completion, and Mathematics. Price \$149.95

EDUCATOR EDITION - includes all of the above programs plus detailed solutions and explanations. Price \$229.95

Independent Tests of S.A.T. series performance show a mean total increase of 70 points in students' scores. Price \$69.95
Update Pack to 81/82 specs. Available to previous owners.

ODYSSEY IN TIME



This spectacular adventure game adds a new dimension of excitement and complexity to **Time Traveler**. Players must now compete with the powerful and treacherous adversary in their exacting quest for victory.

To succeed they must vanquish this adversary in combat that rages across 24 time periods.

Odyssey In Time includes all the challenges of **Time Traveler** plus 10 additional eras, including those of Alexander the Great, Emperor Asoka of India, Attila the Hun, Genghis Khan. Each game is unique, and may be **interrupted and saved** for later play. available for APPLE & TR-80 PET, 32K - \$39.95



TIME TRAVELER

Confronts players with complex decision situations and the demand for real time action. Using the **Time Machine**, players must face a challenging series of environments that include; The Athens of Pericles, Imperial Rome, Nebuchadnezzar's Babylon, Ikhnaton's Egypt, Jerusalem at the time of the crucifixion, The Crusades, Machiavelli's Italy, The French Revolution, The American Revolution, and The English Civil War. Deal with Hitler's Third Reich, Vikings, etc. At the start of each game players may choose a level of difficulty... the more difficult, the greater the time pressure. To succeed you must build alliances and struggle with the ruling powers. Each game is unique.

\$24.95

ISAAC NEWTON



Perhaps the most fascinating and valuable educational game ever devised - ISAAC NEWTON challenges the players to assemble evidence and discern the underlying "Laws of Nature" that have produced this evidence. ISAAC NEWTON is an inductive game that allows players to intervene actively by proposing experiments to determine if new data conform to the "Laws of Nature" in question. Players may set the level of difficulty from simple to fiendishly complex.

In a classroom setting the instructor may elect to choose "Laws of Nature" in accordance with the complete instruction manual provided.

For insight into some of the basic principles underlying ISAAC NEWTON see GODEL, ESCHER, BACH by Douglas R. Hofstadler, Chapter XIX and Martin Gardner's MATHEMATICAL GAMES column in *Scientific American*, October, 1977 and June, 1959. \$24.95.

Send \$2.00 for complete Catalogue.

\$5.00 Discount Coupon included in Catalogue.

PROGRAMS AVAILABLE FOR

TRS-80, APPLE II & PET

(unless otherwise indicated)

disk or cassette (please specify)

All programs require 16K/TRS-80 programs require LEVEL II BASIC/APPLE programs require APPLESOFT BASIC



Krell Software Corp.

Send check or money order to
21 Milbrook Drive, Stony Brook, NY 11790
(516) 751-5139

NY State Residents Add Sales Tax

WHO ARE THE COMPUTER CRIMINALS ?

***A Close Look at Those Who
Are Giving Computer Literacy a Bad Name***

by Jay BloomBecker

The fashion, for journalists and criminologists, is to see all computer criminals as geniuses. Donn Parker, in his book *Crime by Computer*, reflects this when he writes: "Perpetrators are usually bright, eager, highly motivated, courageous, adventuresome, and qualified people willing to accept a technical challenge. They have exactly the characteristics that make them highly desirable employees in data processing." F. W. Dennis, writing in *Security World* in September 1979, paints the same picture: "The common denominator in nearly all cases of computer fraud has been that the individual is very much like the mountain climber — he or she must beat the system because it is there."

But is the picture accurate? Does it help us to prevent computer crime? Based on my analysis of many of the cases in the files of the National Center for Computer Crime Data, I must answer both questions with

a resounding "No."

Many computer criminals just aren't that clever. Their crimes are not that technically sophisticated. And even if it were true, this view of computer crime wouldn't be very helpful. No test has been devised to separate computer geniuses who are prone to crime from law abiding computer geniuses. Lacking this type of test, it wouldn't make much sense to ask a personnel director to screen out all geniuses lest they commit computer crime.

Environment, not personality, seems the most useful factor in predicting computer crime and preventing it. The data suggest that certain "criminogenic environments" are present in most computer crimes. By criminogenic environments I mean the computer system, as the criminal perceives it immediately before he or she decides to commit a computer crime. For example, one type of computer criminal will see the computer environment as a cookie jar — the

source of enough money to meet sudden needs. Another may see it as a playpen — simply a place to play computer games as long as he or she likes. These different perceptions, and the attitudes they reflect, will generally lead to very different types of computer crimes, and require different types of security to prevent them.

In this article, I shall suggest seven views of the computer system which seem to summarize most of the cases that I have seen.

• **The playpen** The American educational television network recently broadcast a one-hour show about computers as tools in the education of nine and ten year olds. About the only discordant note in this paean was one teacher's observation that her students kept erasing each other's names from computer sign-up lists, or destroying the sign-up lists altogether. The teacher tolerated this aberrant behavior, apparently delighted that her students enjoyed play-



An English group calling itself Crank boasts...“computer piracy and security cracking.”

ing at the computer.

Older students have the same morality. Professor John Carroll found that 34 per cent of the students in two courses on advanced information systems at the University of Western Ontario had tried to obtain computer time without paying for it. The same percentage had tried to penetrate the computer's security system.

These examples demonstrate the fact, obvious to anyone who has watched people play computer games, that simply using a computer can be intrinsically satisfying. They also suggest that the drive for this satisfaction can violate other's rights. When computer crime results from the attempt to gain satisfaction from working with the computer, I categorize it as an example of the playpen perception getting out of hand. Unfortunately, there is no standard within the computer industry to define precisely when the playing has got out of hand. Thus, if a student uses an hour of computer time without permission, one university computer department considers it criminal theft of services and another views it as commendable ingenuity.

In addition to the unauthorized use of time (and often as a necessary prerequisite to it) attempts to compromise computer security systems are common to those with a playpen point of view. An English group calling itself Crank boasts of expertise in “computer piracy and security cracking.” Members of Crank claim to have obtained files from universities and companies in England, including one whose security “has never been broken,” all with the ostensible purpose of making sure that computer security is improved. But it would take only a small change in the motivation of Crank members to wind up with someone who seeks to outsmart the computer for his or her own benefit.

This may have been the dominant motivation behind Stanley Rifkin's infamous computer crime. When Rifkin pleaded guilty to charges arising from his theft of \$10.2 million from an American bank, he described his reaction to what he had done. “I was aghast,” he said, explaining that his arrest was the first indication he had had that his scheme really worked. He further explained his failure to hide the diamonds he had bought with the stolen funds, saying he hadn't made any plans for this contingency. Implausible as this story may seem, it is consistent with Rifkin's general ineptitude as a thief and his rapid arrest once he returned to America. Further support for the idea that Rifkin had a playpen perspective comes from an associate who once taught with Rifkin: “The guy is not a bank robber, he's a problem-solver,” was Professor Gerald Smith's opinion.

In view of what I have already said about the playpen attitude towards computer environments, certain security implications immediately become evident. Norms that clearly define allowable and excessive computer use will develop, and if the industry does not develop them, it's quite likely that the law will. In the meantime, each computer user must evaluate

its rules and practices. Unless these clearly communicate the limits on unauthorized computer use, and the rationale for these limits, the user will continue to be vulnerable to the computer “game player” who goes too far.

• **The land of opportunity** Not too far removed from the playpen perspective is the attitude that there's nothing much wrong with exploiting an obvious vulnerability in a computer system. Unlike the game players in their playpen, those who see the computer environment as a “land of opportunity” seem to be motivated not so much by challenge as by lack of challenge. Where the game player might try hundreds of codes to find one which gives entry to the computer's operating system, the employee in the “land of opportunity” just finds and exploits a vulnerability in the system — often in the course of learning his or her job. For example, the operator of a device for printing cheques pressed the REPEAT button again and again when his own cheque was being produced. Hardly the computer genius the media would have us think all computer criminals are, he took a dozen cheques and attempted to cash them all at the same time. More sophisticated, but similarly opportunist, Jack Polak exploited his position as a purchasing agent in the county of San Diego, California. Knowing the troubles the county was having in installing a new computerized system to control its payments for goods bought, Polak set out to compromise the system. He created fictitious vendors, charged the county for non-existent supplies that had ostensibly been delivered, and collected approximately \$50,000 in payments. He knew the county's system too well. Only his impatient questioning about a \$70,000 cheque he awaited led to his detection.

One needn't be a profound criminologist to realize that motivation is directly proportional to opportunity, and opportunity is inversely proportional to the security of a system. If the opportunities are obvious enough, even the least sophisticated employees (or outsiders, for that matter) may be expected to try to exploit them. Consequently, the need for systematic and extensive security cannot be underestimated. Simple measures — such as taking particular care of employees' accounts which are under the computer's control, rotating personnel from job to job, and looking for any trends of irregular work habits which might either give rise to or explain computer crime, are just some of the measures that a company might take to reduce its vulnerability to those who see the computer as a land of opportunity.

• **The cookie jar** A gambling debt, a drug habit, sudden losses on the stock market, may all lead an employee to see the funds in a computer as the best solution to his problem of the moment. As though it were a cookie jar, the “hungry” criminal dips into the computer to take what he needs. In such circumstances, the criminal's motivation is much more pressing than his observation of a loophole in the system's security. Criminologists have identified such “situational pressures” as frequent accompaniments to white collar crime in general; as

A gambling debt, a drug habit, sudden losses on the stock market, may all lead an employee to...the funds in a computer...

computer crime is a subclass of white collar crime, I am not surprised that these pressures operate here as well. Case histories of computer crimes bear me out.

One head teller at a bank in New York City was found to have stolen \$1.5 million when his bookie was raided and it turned out that he was betting up to \$30,000 a day. And in Denver, Colorado, Raymond Ressin financed numerous gambling trips to Las Vegas by falsifying the input to the computer of the stock brokers for whom he worked.

To counteract people whose need drives them to cookie jar crimes, companies must investigate employees before they are hired. But situations change, and security must be responsive to these changes. Ideally, the relationship between employees and their supervisors and management in general will be such that the employee will seek out help when a problem arises.

• **The war zone** Some people see the computer not as a solution to life's problems, but as a symbol of their cause. Disgruntled employees, who feel that management is out to get them or that they have already been hurt unfairly, may express their resentment in attacks on the company computer. This type of employee sees the computer as a “war zone,” the battlefield in the struggle between employer and employee.

In Sacramento, California, three employees of the State Department of Justice, apparently annoyed by the paltry amount of their pay increase, deleted certain arrest records from the State's criminal records. There are numerous other tales of computer library tapes erased, misfiled, or mislabelled, or instructions to erase all company records two years after a certain programmer is fired. Employees have even literally attacked the computer with sharp instruments, screwdrivers, and guns.

In one of the most extreme cases I know of, an employee of a large produce company, feeling cheated out of a substantial pay rise, created a “shadow corporation” for revenge. The shadow was almost exactly like the real corporation, except that it was about 0.75 percent less efficient — more vulnerable to theft, spoilage, and so forth. All of the financial differential between the shadow corporation and the real corporation went into the pockets of the disenchanted employee.

As in the case of the “cookie jar,” good management is the best prevention. Where there are no acceptable avenues to express resentment toward the employer, management should not be surprised when unacceptable avenues are used instead. There is also the possibility that the employee's resentments are well-founded. Corrective action will not only reduce the threat of the resentful employee, but will also demonstrate to all employees the folly of perceiving the computer as a war zone.

• **The soapbox** In the war zone, the criminal gets at the employer through the employer's computer. The soapbox computer criminals find computers themselves anathema, and see

computer crime as a way to strike out *against the computer*. Thus, they see a war going on, not between the employer and themselves, but between themselves and any one of a number of different forces.

In some cases, the slogan of the 1960's — I am a human being; do not fold, staple, or mutilate me — has been converted into an unreasonable dislike for the computer itself. In one case, a programmer sabotaged a computer by short-circuiting the computer's memory. He told the police that he had an “over-powering urge” to shut the computer down, and had no grievance against the owners of the machine.

Terrorist bombings reflect a much more extensively developed symbolism. In Italy, the Red Brigades produced a telling document, *Resolutions of the Strategic Directorate* that depicts increasing computerisation in Western countries as part of a sinister plot to “maximize social controls.” The *Resolutions* explain that because computers are instruments for the “repression of the class struggle . . . it is important to attack, unravel, and dismember these networks of control.”

Where the computer symbolizes personal frustrations, the security measures to deal with the problem are not much different from those called for in the war zone. But soapbox crimes are as likely to be committed by non-employees as by dissidents within. These sorts of attacks on the computer emphasize the need for attention to physical security. The common practice of making the computer center the highlight of a public relations tour is thus a questionable risk, and companies should ensure that they stay aware of the activities of terrorists.

• **The fairyland** If any category of computer criminal completely belies the major myth, it is this one. Some of the participants in computer crime appear to act as though the computer environment were totally divorced from reality. A telex operator may routinely transfer millions of dollars from bank to bank without recognizing how important each transaction is. The computer simply isn't real.

In *Computer Capers*, Thomas Whiteside tells of a fraudulent transfer of \$2 million. The culprit convinced his girlfriend to transfer this amount to his bank in New York, telling her he wanted to play a joke on a computer-operator friend who worked at that bank. The friend and the money disappeared before the girlfriend realized that she had been conned as well as jilted.

Although it is perhaps dangerous to take assertions of ignorance at face value, the message behind cases such as this is quite clear. Where people associated with computer systems are unaware of their own power, and act as though there were no dangers of computer crime, the company who employs them is exposed to an enormous risk. Just as management cannot afford to ignore the new vulnerabilities a computer creates in any business, so it cannot afford to have employees who are ignorant of these vulnerabilities. Clear communication of the responsibilities of the job, as well as clear standards for its per-

Environment...seems the most useful factor in predicting computer crime...

formance are the most obvious management tools to combat this problem; the company should make it clear that each employee has a stake in computer security.

The other side of the fairyland mentality is that even if the employee's ignorance of computing does not facilitate a crime it may stand in the way of that employee detecting the early warning signals which, acted on promptly, might reduce crime. It is clear to me that one of the important security devices in a manual system is that lots of people see each piece of paper, and they can tell unusual entries just because they run counter to common sense. To some extent we lose this security advantage when systems are computerized — and no longer easily understood by many people. If we could only get the game fanatics to teach the residents of fairyland about computing, both might develop more realistic perceptions of the computer environment.

• **The toolbox** To some computer criminals, a computer is simply a computer. Thus, we have cases of so-called computer crime where the computer is not the target, but the implement in a crime that in no other way involves computing. The phone freaks, who use microprocessors to test phone circuits and develop strategies that enable them to use phones for free, exemplify this attitude towards computers. The Equity Funding personnel, who used computers to create false insurance policies by the thousands, seemed similarly clear-headed about what a computer can do. Perhaps the most ludicrous example is the Los Angeles brothel owner who used a minicomputer to keep track of his customers.

These examples suggest a type of computer criminal we can call the technological crook, who sees the computer as another tool in his kit. This type of person is more likely to already be a criminal than any of the individuals whose attitudes I described above. As the key to this sort of crime is computer use, all those strategies which limit access to the computer are important security. But none of these "limited access" security procedures will be at all effective against the owner of a micro- or minicomputer who has worked out a way to use the computer to commit crime.

What To Do Next

The categories listed here are not mutually exclusive, exhaustive, or chiselled in stone, and I encourage feedback — particularly as to attitudes I appear to have missed. Furthermore, the categories may not be much use when we want to predict whether a specific individual is likely to become a criminal (unless we can develop tests to discover if a "suspect" is abnormally likely to perceive the computer environment along the lines the categories suggest). Still, if the company sees the environment as a playpen, a cookie jar, or any of the other models I've discussed, it may be inferred that some would-be computer criminals will see it that way as well. I hope that the security measures I have outlined here will offer some strategies to change or prevent these potentially dangerous perceptions.

Micros Behind Bars "Herman and Me"

by Mike H. Briggs

I have been in prison for the last sixteen years. I still have fifteen years to go. I have a TRS-80 named 'Herman.' We've been together about a year now, and we've become great friends. I've sort of gotten used to the idea of Herman being smarter than me, but then that's the reason I got him in the first place.

I got interested in computers quite by accident. I have a rather large cassette tape collection and reels of music along with a couple hundred LP's. I was looking for a way to index them, and had made several tries with a typewriter, but I didn't like any of them. Then one day I was looking at my new Radio Shack Catalog and saw 'Herman.' Well, it was love at first sight. But how could a prisoner get a computer into this place, much less in his cell? Well, I found out. I just asked the superintendent and the idea was taken before a staff meeting. Finally, I got the approval. But I was still in trouble. First, I didn't have the slightest idea how to run a computer — much less the money to get one. I called the Radio Shack Store and got the complete price along with tax, and then I asked the accounting office how much I had in my account. I found out I had a little more than it would take to purchase a Level I 4K system, so I did.

When Herman got here, I sat down and just looked at him. Then I opened the manual that came with him and was surprised to find I could make sense of it right off. So I plugged Herman in and followed the instructions, and now my system boasts 48K, one disk drive, a speed loader and a voice. (I got a VoiceTek system so Herman and I can talk back and forth.)

Another con and I developed a tape listing program that handles my tapes real well. I also have an information listing program that I modified from one of the many books I bought. I have an address listing that takes the place of my address book, and I have developed a store inventory listing that works with the information that is required to operate a store within a prison. I'm working on a program for the handicapped, which uses endless tapes and allows a person with motor problems to operate a computer with less buttons to push. Also, I have the theory and paperwork done on a program for the blind and handicapped that works with voice applications, both ways: a complete 'talking computer system' that allows a blind person to communicate with the outside world via computer.

I have sent for and received both hardware and software from several companies and have had little or no trouble obtaining them, except for the lack of funds. I earn a wage here but it's rather small, so when I want something I usually have to save for several months before I can send for it.

I am especially interested in the voice applications that a computer can be used for, and with that in mind I've purchased the VoiceTek system. When I first started using it, I found I had trouble with background noises so I installed a CB Mic with a noise cancelling preamp.

Mike Briggs #520850-MSU, P. O. Box 520, Walla Walla, WA 99362.



That solved my problem, except with very loud noises. I'm working on solving that also and I think I have, but I'll have to wait till I have enough funds to send for the items I need. I want to expand my 48K system with one disk drive into at least two drives and a printer so I can work on the voice system I am developing.

I have the time on my hands to develop my knowledge of computers to an extent that I would not be able to on the outside, and I intend to take full advantage of that. I hope to leave here with a good working knowledge of computers and computer systems.

Of all the things I've dabbled in so far, I feel the computer will play a major role in my life. I want to be a part of the future, and I want to be able to do something that will allow me to earn a living and keep out of places like this. I think computers are the answer, and as far as I am able to find out (it's hard to keep up with developments in here) computers are fast becoming a commonplace appliance in home and business. The things you can accomplish with a computer are staggering. I don't think most people have a full idea of the impact of computers on their lives. They will soon be everywhere: in the home, the business, the schools, and I will be a part of that new and exciting world.

One thing I would like to see in this prison would be a school for teaching computer basics. With the hold they are taking on our lives, now would be the time to train people how to use them, and I can't think of a more 'captive' audience than those doing time in prison.

When I first came here in 1966, I didn't have any idea of ever getting out, but now with some skills that will allow me to earn a living and do things that are exciting and useful, my entire outlook has changed. I now know that I can make it, and because of that, I will. But first, I have much to learn, some of it things that people who haven't been exposed to a place like this take for granted. Examples? I went to town one day a few years back to get a tooth fixed and when the car stopped I couldn't get out because I couldn't find the door handle. Another time I went into a store and almost got hit by a door that opened before I got to it. Or the time I went into a bank with a sponsor so he could pay a bill. While standing there waiting, a man asked if he could help me. I said, "No, I'm just browsing." Boy, was I red!

I will have many things to overcome when I face the world again, but now I have the power to do just that. Computers, of course, didn't change my thinking altogether, but they have shown me a way of life that I thought was only for college kids or very smart people.

'Herman' and myself take this time to thank you for your interest. Please feel free to write.

INDEX LISTING 'BASIC' PROGRAM

BY:
STEVEN ENTRINGER & MIKE BRIGGS

This is an indexing program that I use for calling my cassette tapes, either by artist name, by tape #, or by title. Also a default line sends you back to the index if a 'selection is not listed.'

The program is written in Level 2 BASIC. You will need at least 16K or more memory for a healthy index. *See note on bottom of program sheet for memory space.

To see a sample 'RUN' see the last page of this article.

Data lines start at line 320. You will not have to enclose any information within quotes, the program runs very well without them, this makes listing the data easier and much faster and saves on space. Also you don't have to list in any order except of course each listing in itself. Example:

```
320 DATA E. ARNOLD,1,WELCOME,1 & 2
This is one data element within a statement that gives you the four (4) columns to each row thus:
320 DATA E. ARNOLD,1,WELCOME,1 & 2,
E. ARNOLD,2,WELCOME,1 & 2
```

This would produce two rows four columns each. If you make the title too long then you force the side # to drop down a row which in turn forces a space on the screen thus:

```
320 DATA E. ARNOLD,1,WELCOME,1 & 2,
SCHUBERT,45,SYMPHONY # 8 OP.2,1 & 2
When run the 1 & 2 would be down a line and your listing would read:
ARTIST TAPE # TITLE SIDE #
SCHUBERT 45 SYMPHONY # 8 OP.2
1 & 2
```

Therefore if I run into a long title I shorten it so it fits. By running the program using the index listing #2 you can start calling up each tape by the tape number and check each listing before you finish the program listing and data inputting.

'Bout Myself and Steven
=====

I'm in prison. Steven just got released after serving eight years. He is currently on work/release and in school learning computer programming; his class is using TRS-80's. I have had my computer, 'Herman' for about seven months now and have grown really fond of him.

It took me some time to get Level II (funds). Steven wrote the indexing program after only two weeks of studying Levels I & II. Up until that time he had never seen a computer except in articles. So when he asked me to let him try to write a program for me I told him what I needed and this is what he came up with. Steven wrote the program while 'Herman' was still in Level I, and therefore hadn't seen the program work. After I got Level II I ran the program and worked out the bugs. ■

LISTING ON PAGE 22

DISK DRIVE WOES?
 PRINTER INTERACTION?
 MEMORY LOSS?
 ERRATIC OPERATION?

Don't Blame The Software!



Power Line Spikes, Surges & Hash could be the culprit! Floppies, printers, memory & processor often interact! Our unique ISOLATORS eliminate equipment interaction AND curb damaging Power Line Spikes, Surges and Hash.

- ISOLATOR (ISO-1) 3 filter isolated 3-prong sockets; integral Surge/Spike Suppression; 1875 W Maximum load, 1 KW load any socket \$62.95
- ISOLATOR (ISO-2) 2 filter isolated 3-prong socket banks; (6 sockets total); Integral Spike/Surge Suppression; 1875 W Max load, 1 KW either bank \$62.95
- SUPER ISOLATOR (ISO-3), similar to ISO-1 except double filtering & Suppression \$94.95
- ISOLATOR (ISO-4), similar to ISO-1 except unit has 6 Individually filtered sockets \$106.95
- ISOLATOR (ISO-5), similar to ISO-2 except unit has 3 socket banks, 9 sockets total \$87.95
- CIRCUIT BREAKER, any model (add-CB) Add \$ 8.00
- CKT BRKR/SWITCH/PILOT (-CBS) Add \$16.00

Master-Charge, Visa, American Express
 Order Toll Free 1-800-225-4876
 (except AK, HI, MA, PR & Canada)

ESP Electronic Specialists, Inc.

171 South Main Street, Natick, Mass. 01760
 Technical & Non-800-1-617-655-1532

THE ORIGINAL DAN PAYMAR

Lower Case Adapter™

TWO MODELS NOW AVAILABLE

LCA-1™ \$59.95

For revision 6 or older Apple-II* (with RAM configuration blocks).

LCA-2™ \$49.95

For rev. 7 or newer Apple-II* (without RAM configuration blocks).

One of them is designed specifically for your computer.

* No jumpers. The "one board for all Apples" design may require cutting and soldering nine jumpers to adapt it to your computer.

* Only one chip to remove from your computer (two for the LCA-1™). The "one board for all Apples" design requires removing four chips from your computer, then plugging in a complex PC board.

AVAILABLE AT COMPUTER DEALERS EVERYWHERE.

DICE support software
 (diskette) \$5.00 extra.

ENHANCEWARE™ by

MasterCard, Visa,
 and C.O.D. orders
 shipped immediately.

Dan Paymar
 91 Pioneer Place
 Durango, CO 81301
 Telephone (303) 259-3598

Dealer inquiries invited.

*Trademarks of Apple Computer Inc.

"Herman and Me"

HERMAN A TRS-80 LEVEL II 16K+
 TAPE LISTING PROGRAM 7/30/80
 11/10/80

WRITTEN BY STEVEN ENTRIGNER
 EDITED BY M. BRIGGS

```

10 CLS
20 'FILE "A"
30 GOTO 220
40 ?"TO MAKE A REQUEST FOR AN ARTIST:
  TYPE # 1
50 ?"TO MAKE A REQUEST FOR A TAPE # :
  TYPE # 2
60 ?"TO MAKE A REQUEST FOR A TITLE:
  TYPE # 3
70 ?"TO MAKE A NEW LISTING GOTO 320 OR
  BEYOND:
80 ?"===== "
90 Z=0: INPUT"TYPE #";X: ON X GOTO 100,130,
  160
100 INPUT"ARTIST NAME";NS: GOTO 210
110 FOR R = 1 TO N: IF A$(R,1)=NS GOTO 220
120 NEXT: IF Z>0 THEN 200 ELSE 190
130 INPUT"TAPE #";TS: GOTO 210
140 FOR R = 1 TO N: IF A$(R,2)=TS GOTO 220
150 NEXT: IF Z>0 THEN 200 ELSE 190
160 INPUT"ALBUM TITLE";BS: GOTO 210
170 FOR R = 1 TO N: IF A$(R,3)=BS GOTO 220
180 NEXT: IF Z>0 THEN 200 ELSE 190
190 ?"SELECTION IS NOT LISTED": FOR D=1 TO
  450: NEXT: CLS: GOTO 40
200 INPUT"IF YOU WISH TO SEE THE INDEX
  AGAIN PRESS 'ENTER' KEY";X:CLS:GOTO 40
210 ?"ARTIST","TAPE #","TITLE",
  TAB(58)A$(R,4):Z=Z+1:ON X GOTO 110,140,
  170
220 ?A$(R,1),A$(R,2),A$(R,3),TAB(58)A$(R,4):
  Z=Z+1:ON X GOTO 120,150,180
  
```

SUBROUTINE

```

230 ?
240 CLEAR
250 N=Y          'Y MUST EQUAL NUMBER OF
                DATA STATEMENTS

260 DIM A$(N,4)
270 FOR R = 1 TO N
280 FOR C = 1 TO 4
290 READ A$(R,C)
300 NEXT C,R
310 GOTO 40: CLS
320 DATA LINES START HERE
  
```

16K WILL LIST 194 TAPES LEAVING ENOUGH
 MEMORY FOR STRINGS AND DATA

"Computers, of course, didn't change my thinking altogether, but they have shown me a way of life that I thought was only for college kids or very smart people."

SAMPLE 'RUN' OF TAPE LISTING
 PROGRAM

BY:
 STEVEN ENTRINGER
 MIKE BRIGGS
 11/10/80

```

RUN
TO MAKE A REQUEST FOR AN ARTIST: TYPE # 1
TO MAKE A REQUEST FOR A TAPE # : TYPE # 2
TO MAKE A REQUEST FOR A TITLE : TYPE # 3
TO MAKE A NEW LISTING GOTO 320 OR BEYOND :
  
```

```

=====
TYPE #?1
ARTIST'S NAME? E. ARNOLD
ARTIST      TAPE #      TITLE      SIDE #
E. ARNOLD   1          WELCOME    1 & 2
E. ARNOLD   2          WELCOME    1 & 2
E. ARNOLD   3          WELCOME    1 & 2
IF YOU WISH TO SEE THE INDEX AGAIN, PRESS 'ENTER' KEY?
  
```

```

RUN
TO MAKE A REQUEST FOR AN ARTIST: TYPE # 1
TO MAKE A REQUEST FOR A TAPE # : TYPE # 2
TO MAKE A REQUEST FOR A TITLE : TYPE # 3
TO MAKE A NEW LISTING GOTO 320 OR BEYOND :
  
```

```

=====
TYPE #?2
TAPE #?15
ARTIST      TAPE #      TITLE      SIDE #
E. FITZGERALD 15          COLE PORTER 1 & 2
IF YOU WISH TO SEE THE INDEX AGAIN, PRESS 'ENTER' KEY?
  
```

```

RUN
TO MAKE A REQUEST FOR AN ARTIST: TYPE # 1
TO MAKE A REQUEST FOR A TAPE # : TYPE # 2
TO MAKE A REQUEST FOR A TITLE : TYPE # 3
TO MAKE A NEW LISTING GOTO 320 OR BEYOND :
  
```

```

=====
TYPE #?3
ALBUM TITLE? WILD IS LOVE
ARTIST      TAPE #      TITLE      SIDE #
KING COLE   30          WILD IS LOVE 1
IF YOU WISH TO SEE THE INDEX AGAIN, PRESS 'ENTER' KEY?
  
```

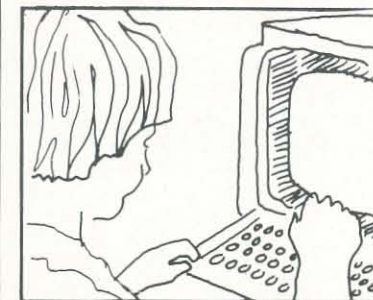
```

RUN
TO MAKE A REQUEST FOR AN ARTIST: TYPE # 1
TO MAKE A REQUEST FOR A TAPE # : TYPE # 2
TO MAKE A REQUEST FOR A TITLE : TYPE # 3
TO MAKE A NEW LISTING GOTO 320 OR BEYOND :
  
```

```

=====
TYPE #?2
TAPE # 40
ARTIST      TAPE #      TITLE      SIDE #
SELECTION IS NOT LISTED.
                (THIS IS ON A TIMER LOOP IN LINE 190
                WHICH RETURNS YOU TO THE INDEX.)
  
```

Coming Soon!
 A CHILDREN'S REVOLUTION
 DON'T BOTHER ME,
 I'M LEARNING



See and listen to children learning on a computer in this motivating 24 minute film on the use of the computer in the classroom. DBMIL is an excellent introduction into the future and benefits of computer assisted education.

For more information on how you can preview this important innovation in education write:

CRM/McGRAW HILL
 110 Fifteenth St.
 Del Mar, Ca. 92014
 714-453-5000

! NEW !

Classified Ad Space Available

ONLY \$5 per 40 character line including spaces & punctuation. 3 line minimum. Write Advertising Dept., Recreational Computing, P. O. Box E, Menlo Park, CA 94025.

AVAILABLE NOW! THE INTERNATIONAL MICROCOMPUTER SOFTWARE DIRECTORY

ONLY \$28.95 + \$2.95 post.

5000 entries, cross-referenced by machine, O/S, application, vendor and program name.

Phone or write for details. All major credit cards accepted.

IMPRINT SOFTWARE
 US - 420 South Howes St., Ft. Collins, CO
 80521 - Tel 303 482 5574
 UK - 16 Milton Avenue, Highgate, London N6
 -Tel 01 348 3998

How to Start a Software Exchange

by Ann Lathrop and Bobby Goodson

"What is different about SOFTSWAP? Isn't it just another users' group?"

Recipe for a SOFTSWAP

Take one enthusiastic, experienced group of COMPUTER-USING EDUCATORS,

Add one COUNTY OFFICE OF EDUCATION with an established reputation for innovative leadership,

Stir in a collection of PUBLIC DOMAIN EDUCATIONAL SOFTWARE contributed by CUE members,

Season with generous loans of MICROCOMPUTERS from Commodore, Radio Shack, APPLE, CompuColor, and Bell & Howell,

Combine all ingredients with one LIBRARIAN who has strong organizational skills and a firm commitment to educational uses of microcomputers,

Sprinkle this rich mixture with many hours of CUE MEMBERS' VOLUNTEER WORK,

Let marinate for about ONE YEAR.

Recipe yields one SOFTSWAP . . . freely and gladly shared with Computer-Using Educators everywhere!

Ann Lathrop is Library Coordinator at the San Mateo County Office of Education, Redwood City, CA.

Bobby Goodson is President of Computer-Using Educators, and Computer Resource Specialist at the Cupertino Union School District, Cupertino, CA.



Visitors to Softswap receive assistance from Microcomputer Center workers and from one another.

The Microcomputer Center in Redwood City, California, is a gathering place for members of Computer-Using Educators (CUE) and visitors from other areas who enjoy sharing their ideas, problems and expertise in this unique resource center. It's an exciting place to spend an afternoon, a few days, or longer, and the welcome mat is always out for educators who want to help with various CUE projects, to copy SOFTSWAP programs, or who simply like to sit and talk with other microcomputer enthusiasts who happen to drop in. Questions and answers fly back and forth, and frequent arguments as to which answer is correct reflect the diversity of opinion and experience represented by the group. At any one time the questions may vary from "What computer is best?" and "Can I do the school budget and attendance on this one?" to "How can I get this program to run off of our Corvus?" and "What is the difference between 3.2 and 3.3 DOS?"

Many answers are derived from do-it-yourself experience with the microcomputer systems, commercial packages, and SOFTSWAP programs which are available to visitors. Other answers are found by consulting the resource file of names and addresses of people experienced in almost all aspects of microcomputing. The books, journals and extensive reference files in the library provide still more answers to visitors' questions. The people in the Center, both staff and visitors, are also a valuable source of information and expertise.

The Microcomputer Center was established just over a year ago in the Library of the San Mateo Educational Resources Center (SMERC) as a joint project of Computer-Using Educators and the San Mateo County Office of Education. Visitors have come from throughout California, from many other states

and Canadian provinces, and from Australia and France. All have been interested in learning more about the challenging potential of microcomputers as new instructional media. By supporting the establishment of this unusual educational resource, the San Mateo County Office of Education, under the leadership of Superintendent William K. Jennings, continues its traditional role of developing innovative educational programs in California.

There has been no outside funding of the Microcomputer Center. The San Mateo County Office of Education provides the space, maintenance and part-time staff support as a service to educators in the county and throughout the San Francisco Bay area. The active involvement of Computer-Using Educators as co-sponsor of the Center has been the other major factor in its success.

CUE has grown into an organization of over 1600 educators, with members in 40 states, 12 provinces and 12 foreign countries. CUE members donate hundreds of hours of volunteer time to organize and participate in activities of the center, including work on SOFTSWAP programs, demonstrations of new equipment and software, and the commercial software evaluation project.

Many members of the county office staff cooperated in the development of the Microcomputer Center and continue to be active in its operation. The original concept was presented by Ann Lathrop, Library Coordinator, who continues to supervise the SOFTSWAP and the software evaluation project. Assistance and support for the project was given by Dr. Curtis May, Director of Library Services, whose commitment has been essential to the successful establishment of the center. Visitors receive friendly help from Janice Marshall, Library Assistant, and from her staff of student aides. Curricu-

lum development programs and in-service training are the responsibility of LeRoy Finkel, computer consultant, and of Walter Smithey, math-science consultant, working under the direction of Tom Quinn, Director of Curriculum. Dr. James Fee, Director of Media Services, offers assistance with hardware applications and demonstrations. An extensive program to interface computers and microcomputers throughout the county's schools is being developed by Dr. Al Grossman, Administrator of the Office of Information Services. The entire project is under the administration of Dr. Don Halverson, Assistant Superintendent for Educational Support and Planning. This team approach by the county office staff, coupled with the expertise and commitment of Computer-Using Educators, has been crucial to the successful development of the Microcomputer Center.

Manufacturers of microcomputer hardware have also been generous in their support of the center. Tandy (Radio Shack) and Commodore (PET) provided the first microcomputer systems and thus initiated the support that has made possible all subsequent development. Each firm set up a complete system with microcomputer, dual-disk drives, printer and cassette-recorder, all on long-term loan of the Microcomputer Center. A library of commercial programs loaned to the center by each firm includes educational software and word processing systems.

Other firms providing equipment for the center including APPLE, Atari, Bell & Howell and CompuColor. All of the systems are on long-term loan; the center has not purchased any hardware. CUE members were of great assistance in contacting the various firms and securing their support and cooperation.

All of the manufacturers have been helpful in maintaining the equipment in good condition, despite heavy use in the center. They also provide new models as they are developed. This strong commitment on the part of the manufacturers has been another critical factor in the success of the center.

Development of the SOFTSWAP

The SOFTSWAP began as an exchange of instructional programs at the Asilomar Math Conference in 1979. Vince Contreras, San Jose State University professor, organized this first attempt. The programs contributed by participants were deposited in the SMERC Library in the spring of 1980. The first of the equipment being loaned to the center arrived from the manufacturers just as school was out. Interested educators worked throughout the summer to evaluate, edit and revise the programs from the Asilomar Conference.

CUE members served as an Advisory Committee for the Microcomputer Center

and also supervised the processing of SOFTSWAP programs for each computer. The committee consisted of Jim Love (PET), Brian Sakai (TRS-80), Dave Stone and Bob Enenstein (APPLE), Marge Fitting (CompuColor), and Pat Tubbs (Atari). The chairpersons also conducted informal demonstrations and evaluation sessions during the summer. As we enter our second year of the SOFTSWAP project, Leslie Grimm is the new chairperson for APPLE; the other systems continue under the direction of the original Advisory Committee members.

Public support of the SOFTSWAP is encouraging. A large number of volunteers continue to donate programs, time and expertise to this growing collection. As new programs flow into the center they are routed to the CUE volunteers for evaluation, debugging and processing. Finished programs are placed onto SOFTSWAP dissemination disks which can be copied without charge by anyone who visits the center.

The large number of requests from outside the San Francisco Bay area led to the mail-order policies adopted this spring. The first catalog was published in the March 1981 CUE Newsletter and listed over 200 programs on 12 disks. Four additional disks have been completed, bringing the total number of programs in the SOFTSWAP to nearly 250. Anyone who sends an original program on a diskette as a contribution to the SOFTSWAP can request a free disk of programs in exchange. Disks are also sold for the cost of the disk, handling and postage. During the first three months of operation over 370 disks have been sent out, some going as far away as Buenos Aires, Singapore and Paris.

SOFTSWAP Procedures

What is different about the SOFTSWAP? Isn't it just another users' group? It is a users' group, but the primary distinction is one of focus - SOFTSWAP concentrates on the development of educational software. Most programs are short, stand-alone instructional units. Many are drill-and-practice exercises written for the elementary school or for remedial work at the secondary level. About one-third of the programs are math oriented. Most games have been excluded since these are generally available from other sources. Only a few utilities, those of special interest to teachers, were on the first series of SOFTSWAP disks; these also are readily available from many user groups.

Every program is evaluated by at least two educators before it is added to a dissemination disk and made available for copying. Programs are edited for spelling errors, inaccurate or incomplete instructions, errors in factual content, programming errors and other problems.

Our goal is to distribute only those programs which have some educational value and which are as free of errors as possible in both content and programming. The actual quality of the programs in the collection varies greatly, as does program length. Each disk contains from 12 to 30 programs, all for one system, but includes various subjects and grade levels.

Donors complete a Preliminary Inventory Form that identifies and describes the program, and a Release Form that gives CUE permission to distribute the program. The donor's form also asks whether the program is original, a modification of another program, or an adaptation from a magazine or book listing.*

The contributed program is put onto a receiving disk where it will remain in its original contributed form as a permanent record. Filled receiving disks are placed into our permanent archives after all of the programs on that disk have been processed. One donated program at a time is loaded from the receiving disk onto a working disk and all subsequent versions of that program will be stored on the same disk until the program is finished and ready to be transferred onto the dissemination disk. We have approximately 40 working disks, each with only one program, being evaluated at any given time. At least two educators will review the program on the working disk and make needed revisions. When they agree that it is finished the program will be removed from the working disk and added to the current dissemination disk in the SOFTSWAP. Another program from the receiving disk is then loaded onto that working disk and the evaluation process is repeated. All finished programs on the dissemination disks are available for copying by visitors.

Two forms are used in the evaluation of program: Checklist for Microcomputer Program Revision and Programmer Revision Sheet. These forms list common errors and problem areas which evaluators are to check. Detailed notes describing needed revisions are made on the Programmer sheet and the final reviewer checks to see that all requested corrections have been made. The finished program is described on a final Inventory Form - SOFTSWAP and then added to the SOFTSWAP catalog. Newly developed programs are also listed periodically in the CUE Newsletter.

These evaluation forms are not designed to produce critical reviews for publication but to provide a guide to be followed in polishing the programs for classroom use. Programs are checked for accuracy of content, clear instructions, trapping of input errors, etc. Certain problems which frequently appear in con-

*Samples of the forms used for the SOFTSWAP are available from the authors at the San Mateo County Office of Education, 333 Main St., Redwood City, CA 94063.

tributed software could be reduced or eliminated if donors would check the following:

- 1) Does the program run with relatively few problems? (Unfortunately, we receive many programs which do not load/run at all.)
- 2) Does the program trap input errors? Does it crash if the user inputs numeric responses when the computer is expecting alpha, or vice versa? What happens if the user hits an unwanted ENTER/RETURN? When there is an input error, is the computer response clear and simple? "Input error, redo from the start" is not helpful to the novice.
- 3) Are all the words spelled correctly?
- 4) Are the instructions and other information easy to read on the screen? Is it formatted to avoid split words at the end of a line? It is best to avoid crowding and to clear the screen at appropriate intervals. Instructions should be written at a reading level appropriate for the intended user.
- 5) Is the *positive* reinforcement for the correct responses more interesting than the negative reinforcement for errors? Graphics in most "Hangman" type games, for example, are usually more exciting when the user loses and is "hanged." Students will often deliberately miss a problem or a word in order to see a rocket explode, a boat sink, etc. It is best to try to provide strong encouragement for doing a task correctly and yet avoid over-praising. It is also better to provide several responses for both correct and incorrect answers - a little variety adds a great deal to the program.
- 6) Have you included REM statements as needed? They make it easier to use a program effectively. It is especially helpful to include statements telling the teacher how to modify word lists, problems or other entries.
- 7) Do you make negative remarks to or about the student? Sarcastic statements can be discouraging. We remove comments like "Boy was that dumb," "Stupid," "Dummy," etc.
- 8) Do you provide hints and/or eventually give the user the correct answer? We usually allow only 3 or 4 incorrect responses before providing either a hint or the correct response, often with some instruction or explanation. Exceptions are programs which are tests or timed drills for accuracy.
- 9) Is the content accurate? Are the definitions and/or explanations simple and straightforward? Do they correspond to the material in the more frequently used texts?
- 10) If it is a drill-and-practice program, do examples come up in random order? Are there enough problems or questions that the student will not have all the same examples if they run the program again? If it is appropriate, does the student have the option to return to the beginning of the exercise and try it again?

Finally, we want to be sure that the program submitted is truly an original creation of the author. If it is an adaptation from another program, from a book or a journal we want to be aware of that fact. We will request the author's permission to include the program in our collection. If we discover that a SOFTSWAP program is actually an infringement on someone else's copyright, we remove it from the dissemination disk.

The most exciting programs we receive are the creative ones that do more than just put workbook pages and drill problems on the screen. Contributed software is improving as educators become more sophisticated users and programmers. Newer programs are often more challenging to the student user and make better use of the unique capabilities of the computer. We are especially pleased when we receive donations of this type and take special pride in being able to make them available to other teachers through the SOFTSWAP.

Evaluation of Commercial Software

The newest project in the Microcomputer Center is the development of an exemplary collection of commercially produced educational software. We have contacted publishers of microcomputer software with requests for review copies of their educational packages. To date the response has been very encouraging. Many publishers appear to welcome the opportunity to put their materials on display for evaluation. CUE members are previewing and evaluating this software during the summer and the project will continue throughout the coming school year, with tentative plans to publish a collection of reviews. Visitors to the center

may preview the software and are invited to help with the review process. However, this software may *not* be copied.

Our goal is to make selected software available for each microcomputer system in the center in order to demonstrate the special features of the system. We also want to give the potential user some idea of the range and quality of materials being published in various subject areas. Finally, we will provide the opportunity for educators to preview a wide variety of software before making purchase decisions. Since it is currently very difficult to find reliable critical reviews of computer software, and because many teachers prefer to make their own evaluation before purchase, we believe that this will be an especially valuable service.

We are also developing a collection of software to demonstrate administrative applications of microcomputers. We will preview and display software designed for school administrators, including word processors and data base management programs. We are also especially interested in potential uses of this type of software in special education programs.

As part of our commitment to encouraging the development of good critical evaluations of microcomputer software, the Microcomputer Center is a participating member of MicroSIFT, a Northwest Regional Educational Laboratory project funded by the National Institute of Education. The objective is to produce and disseminate a high-quality evaluation instrument and to publish reviews of computer software evaluated in terms of the criteria developed. CUE members, under the coordination of the Microcomputer Center staff, have already completed the first round of MicroSIFT program reviews.

Summary

Any new project is accompanied by often unexpected and urgent needs. More than the usual number of critical needs seem to surface with an educational microcomputer project - perhaps because there is apt to be a lack of readily available support resources. The partnership of CUE and the San Mateo County Office of Education has created the Microcomputer Center to meet many of these needs. We have provided educators with a place to see and try various microcomputer systems where other educators who understand their concerns are available. We have created a demonstration site for commercial software, where teachers can "try before they buy," and where they can develop the ability to do critical and objective evaluation of software. We are providing free, educational software and a way for teachers to exchange their own programs with others. We hope we have created a model that will work in other areas and thus will play an even more significant role in developing uses of microcomputers in education. ■



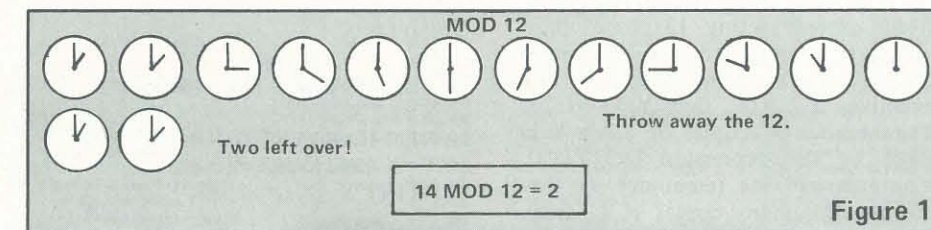
Bobby Goodson, one of the moving forces of Computer-Using Educators, possesses a wealth of information about educational software.

Some Solutions on page 28.

Modular Arithmetic or How to Cook with Leftovers

Leftovers are interesting, useful and valuable things. The leftover Thanksgiving turkey makes tasty sandwiches. Leftover money makes a nice savings account. Leftover string makes a nice ball. Leftover rags make nice rugs. Leftover numbers make nice arithmetic.

The arithmetic of leftover numbers is called *modular arithmetic*. The theory of leftovers is a highly developed science. You have used it many times. Clocks use leftover arithmetic.



If you start a job at noon and it takes 14 hours, then what time will it be when you finish? The clock has only 12 numbers. The leftover is $14 - 12 = 2$. You will finish at 2 o'clock. Of course, there are 24-hour clocks. Those repeat after 24 hours. Short-wave radio and sea-going people use this system. You are using a MOD 12 modular arithmetic system to tell time by a 12-hour clock. MOD 24 would be used with a 24-hour clock.

Leftovers can cause serious social and political problems. Consider what happens when 7 serious gum chewers try to divide 54 sticks of gum. The 5 leftover pieces are a definite problem. Wars have been fought over less.

Leftovers are everywhere. You may have heard the story about the computer programmer at a certain well-known bank who noticed that numbers like \$3.254 often occur. The observant programmer noticed that banks worked only to the nearest cent and never in fractions of a cent. He wrote a program to put the leftovers in his own account. The poor fellow was caught only because the leftovers grew so large that his account attracted attention.

Gum Runners

The ComputerTown Gum and Social Society gathers on the third Wednesday of every other month for a tasting party. The members of the society taste and evaluate the finest of the world's gum. The leftover gum has occasionally been a problem since it is so hard to divide evenly.

If the chairperson of the society knew beforehand how many leftovers there would be, then precautions could be taken to avoid loud arguments over the last few pieces. What is needed is a computer program that takes as input the number X of pieces of gum, the number Y of members present, and then prints out the number of leftover pieces.

Hint for Gum Runners

Well, pass out the gum. Let's see, how many pieces of gum are there? Looks like the number of pieces of gum this time is $X = 23$. (We just made that up.) How many gum chewers are there? It

looks like $Y = 7$ gum chewers. (we made that up, too.) Let's pass out the gum. One for you, and one for you, and . . . We can keep passing out gum in groups of seven until there is not enough to pass out another group of seven.

Germ Harvest

The ComputerTown College Biology Department has an experiment underway. They are growing amoebas. You've probably seen amoebas under a microscope. They're squishy, round, little devils. Amoebas make other amoebas in the most simple and unromantic way. Each amoeba splits itself into two amoebas. This happens quite regularly for the amoebas in the experiment. In fact, it happens every hour, on the hour.

The amoebas are harvested occasionally, right after they have split. The harvested amoebas are divided equally among 13 different bottles. The harvested amoebas are not allowed to split anymore (we guess that the Biology Department knows how to remove the splitter from amoebas). Any leftover amoebas are allowed to stay and keep on growing.

At first there was only one amoeba. At the end of 1 hour, there were two

amoebas. At the end of 2 hours, there were 4 amoebas. At the end of 3 hours, there were 8. At the end of 4 hours, there were 16. This was enough to harvest, so 13 amoebas were put away. There were only 3 left. After 5 hours the 3 had grown into 6, and so on.

The Biology Department would like a computer program that prints out the number of remaining amoebas at the end of each hour. If a harvest has taken place, the number to be printed out is the number left after the harvest.

Hint to Germ Harvest

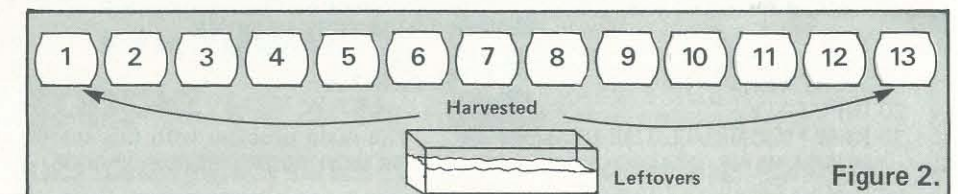
Those tiny devils keep splitting in two. That makes twice as many. Whatever is left after the harvest will double during the next period.

- TIME 0 LET Y=1. This is the initial number of amoebas. $Y < 13$ so there are not enough to harvest.
- TIME 1 $Y = 2 * Y = 2$. Y has doubled, but $Y < 13$. There are not enough to harvest yet.
- TIME 2 $Y = 2 * Y = 4$. Y has doubled again. $Y < 13$. There are still not enough to harvest.
- TIME 3 $Y = 2 * Y = 8$. $Y < 13$. No harvest.
- TIME 4 $Y = 2 * Y = 16$. $Y \geq 13$. There are now enough to harvest. Take away groups of 13 as many times as possible. $Y = Y - 13 = 3$. Since $Y < 13$ now, no more can be harvested.
- TIME 5 $Y = 2 * Y = 6$. $Y < 13$. No harvest. etc., etc. (Solutions on page 28.)

Some Old Business

Ed Coudal sent a nice solution to the reversal problem in the July/August column. The problem was to take a 20-character line, and invert it. His solution is written in North Star BASIC which uses the '+' as a concatenation operator. Here is his solution.

```
100 DIM A$(20), A1$(21)
105 A1$=""
110 INPUT A$
120 FOR I=20 TO 1 STEP -1 Pick off
130 B$=A$(I,I) the tail of A$
140 A1$=A1$+B$ Glue it onto A1$
150 NEXT
170 !A1$ Print A1$
```



Solutions to Programming Problems

Solution to Gum Runners #1

One way to solve this problem is to pass out the gum just like 5-year olds do. Just keep going around the circle until there's not enough gum to go around again. A program to do this:

```
10 X=23
20 Y=7
30 REM **TEST IF THERE IS
  ENOUGH FOR Y PIECES**
40 IF X < Y THEN 110
50 REM **PASS OUT Y PIECES -
  DECREASE X**
60 FOR W=1 TO Y
70 X=X-1
80 NEXT W
90 REM **GO CHECK IF DONE**
100 GOTO 40
110 PRINT X
120 END
```

This program has some good points and some bad points. The good point is that it follows closely the way that 5-year olds pass out gum. The bad points are many. First of all, lines 60, 70, and 80 do nothing but subtract Y from X. We could replace lines 60-80 with a single line:

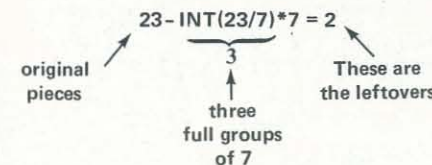
60 X=X-Y

This would pass out Y pieces all at once. The program would run much faster with this change. It would still be quite slow when X is a big number and Y is small. In such a case, it might be necessary to do thousands of subtractions in line 60. Your computer is fast, but such things do take time.

Here is another approach which is based on division.

Solution to Gum Runners #2

The quick way to find out how many 7's there are in 23 is to divide 23 by 7 getting 3.28574286. The integer part of this division is 3. This tells us that there are three 7's in 23 with a bit left over. To find out how much is left, we subtract 3*7 from 23. There are 2 remaining pieces. In BASIC, the computation is performed by:



The program for this second solution:

```
10 REM **X PIECES OF GUM AND
  Y CHEWERS**
20 INPUT X,Y
30 REM **COMPUTE THE REMAIN-
  DER**
```

```
40 Z=X-INT(X/Y)*Y
50 REM **PRINT REMAINDER**
60 PRINT Z
```

Solution to Germ Harvest #1

```
10 REM **Y IS NUMBER OF
  AMOEBAS**
20 Y=1
30 REM **Y DOUBLES**
40 Y=2*Y
50 REM **GO PRINT IF NO
  HARVEST**
60 IF Y < 13 THEN 100
70 REM **HARVEST**
80 Y=Y-13
90 GOTO 60
100 PRINT Y
110 GOTO 40
```

Once again, the first solution can be improved by performing a division instead of subtracting 13 many times. Lines 50-90 need to be changed. Instead of subtracting 13 some number of times you can find how many copies of 13 are contained in Y. For example, if Y=16, then $Y/13 \approx 1.23$. The number of copies of 13 in Y is given by the expression $INT(Y/13)$. You can get the remainder by subtracting this many copies of 13 from Y. The remainder is $Y-INT(Y/13)*13$.

Changes for Solution to Germ Harvest #2

```
50 REM **HARVEST**
60 Y=Y-INT(Y/13)*13
70 ← delete line 70
80 PRINT Y
90 GOTO 40
100 ← delete lines 100
110 ← and 110
```

Now that you have seen these two solutions, can you write a program that allows the user to input the number of hours X that have gone by and then prints the number of amoebas remaining at that particular time?

The most straightforward way to do this is to add a counter to the previous program and print out the value of Y when enough hours have gone by.

Solution for Germ Harvest #3

```
10 Y=1: REM Y IS INITIAL AMOEBA
  NUMBER
20 Z=0: REM Z IS THE COUNTER
30 INPUT X: REM X IS NUMBER OF
  HOURS
40 Y=2*Y: REM Y DOUBLES
50 Y=Y-INT(Y/13)*13
60 Z=Z+1: REM INCREMENT HOUR
  COUNTER
70 REM TEST IF DONE
80 IF Z < X THEN 40
90 PRINT Z
```

The main problem with this solution is that if X is big, say X=1000,

then the computations in lines 40-60 will be done 1000 times. It takes some time to perform each BASIC statement. An estimation to perform lines 40-80 could be in the order of a second. A thousand seconds is nearly 17 minutes. That is too long to wait!

Observe what happens if we let the amoebas grow awhile before we harvest the excess. If the amoebas grow for 5 hours, there will be $2 \uparrow 5 = 32$ of the little rascals. If we now harvest groups of 13, we are left with 6. This is just the amount we would have had if we had harvested at each step. We have avoided all those in-between steps. That should speed things up.

Here is a program that attempts to use this technique. The logic is OK, but there is a practical difficulty. Can you see why this perfectly good program will not work?

```
10 INPUT X
20 Y=1
30 REM DOUBLE Y,X TIMES
40 Y=2 \uparrow X
50 REM HARVEST GROUPS OF 13
60 Y=Y-INT(Y/13)*13
70 PRINT Y
```

Suppose X=1000. How big will $2 \uparrow X$ be? Will any computer be able to handle such a number? It is not very likely since $2 \uparrow 1000$ is about equal to $10 \uparrow 301$. That's a 1 followed by 301 zeros.

We can salvage something from the last approach. The computer can handle $2 \uparrow 16 = 65536$ without any difficulty. We could break X=1000 up into chunks of 16.

The program will look like this:

```
10 X=1000: REM X COUNTS THE
  HOURS
20 Y=1: REM Y WILL COUNT
  AMOEBAS
30 REM *SET THE NUMBER Z OF
  GROWTH PERIODS*
40 Z=16
50 IF X < 16 LET Z=X
60 REM LET Y GROW FOR Z
  HOURS
70 Y=2 \uparrow Z *Y
80 REM HARVEST THE EXCESS
90 Y=Y-INT(Y/13)*13
100 REM ROUND Y TO INTEGER
110 Y=INT(Y+.5)
120 REM DECREMENT THE HOUR
  COUNTED
130 X=X-Z
140 REM CHECK IF DONE
150 IF X > 0 THEN 40
160 PRINT Y
```

Yet Another Method to Harvest Germs

There is one last method that uses an important and famous bit of number trickery. Pierre Fermat discovered

HUNTINGTON COMPUTING

PROGRAMS LISTED BELOW ARE ON DISK:

Now Selling Atari, PET, TRS-80 Software

Atari		Pet		TRS-80	
ADVENTURE INTERNATIONAL					
Adventureland (cass.)	\$14.95 now	\$12.74			
Pirate's Adventure (cass.)	\$14.95 now	\$12.74			
Mission Impossible (cass.)	\$14.95 now	\$12.74			
Voodoo Castle (cass.)	\$14.95 now	\$12.74			
The Count (cass.)	\$14.95 now	\$12.74			
Strange Odyssey (cass.)	\$14.95 now	\$12.74			
Mystery Fun House (cass.)	\$14.95 now	\$12.74			
Pyramid of Doom (cass.)	\$14.95 now	\$12.74			
Ghost Town (cass.)	\$14.95 now	\$12.74			
Savage Island I (cass.)	\$14.95 now	\$12.74			
Savage Island II (cass.)	\$14.95 now	\$12.74			
Angle Worms/Croton Div. (cass.)	\$9.95 now	\$8.54			
Deflection (cass.)	\$9.95 now	\$8.54			
Mountain Shoot (cass.)	\$9.95 now	\$8.54			
Sunday Golf (cass.)	\$9.95 now	\$8.54			
Galactic Empire (cass.)	\$19.95 now	\$16.94			
Star Trek 3 5 (cass.)	\$14.95 now	\$12.74			
Lunar Lander (cass.)	\$14.95 now	\$12.74			
ARTSCI					
Poker Solitaire (cass.)	\$14.95 now	\$12.74			
Gomoku (cass.)	\$19.95 now	\$16.94			
Reversi (cass.)	\$19.95 now	\$16.94			
Cypher Bowl (cass.)	\$29.95 now	\$25.44			
AUTOMATED SIMULATIONS					
Rescue at Rigel (cass.)	\$29.95 now	\$25.44			
Star Warrior (cass.)	\$39.95 now	\$33.94			
Invasion Orion (cass.)	\$24.95 now	\$21.24			
Datestones of Ryn (cass.)	\$19.95 now	\$16.94			
AVALON HILL GAME COMPANY					
Empire of the Overmind (disk)	\$35.00 now	\$29.74			
Conflict 2500 (cass.)	\$15.00 now	\$12.74			
Empire of the Overmind (cass.)	\$30.00 now	\$33.94			
Tanktics (cass.)	\$24.00 now	\$20.44			
DATASOFT					
Atari Mailing List (disk)	\$24.95 now	\$21.24			
Atari Character Generator (disk)	\$19.95 now	\$16.94			
Text Wizard (disk)	\$99.95 now	\$84.44			
Atari Character Gen. (cass.)	\$15.95 now	\$13.54			
Le Stick	\$39.95 now	\$33.94			
ON-LINE SYSTEMS					
HI-RES Adv #2 - Wiz & Princess (disk)	\$32.95 now	\$27.94			
PERSONAL SOFTWARE					
VisiCalc (disk)	\$199.95 now	\$164.44			
Checker King (cass.)	\$19.95 now	\$16.94			
MicroChess (cass.)	\$19.95 now	\$16.94			
UNITED SOFTWARE OF AMERICA					
Survival/Adventure (disk)	\$24.95 now	\$21.24			
3-D Supergraphics (disk)	\$39.95 now	\$33.94			
3-D Supergraphics (cass.)	\$39.95 now	\$33.94			
VERSA COMPUTING					
Mind-Bogglers I (disk)	\$19.95 now	\$16.94			
Mind-Bogglers I (cass.)	\$15.95 now	\$13.54			
VersaWriter Graphics Tablet	\$300.00 now	\$254.44			
Hidden Words	\$17.50 now	\$15.44			
Spatial Relations	\$17.50 now	\$15.44			
Word-Scramble	\$15.00 now	\$13.54			
Preschool Fun	\$15.00 now	\$13.54			
Fastgammon (cass.)	\$19.95 now	\$16.94			
Assembler (cass.)	\$24.95 now	\$21.14			
6502 Disassembler (cass.)	\$11.95 now	\$10.14			
6502 Disassembler (disk)	\$11.95 now	\$10.14			
Tank Trap (cass.)	\$11.95 now	\$10.14			
Tank Trap (disk)	\$14.95 now	\$12.64			
Tari Trek (cass.)	\$11.95 now	\$10.14			
QS Forth (disk)	\$79.95 now	\$67.94			
OS Base Hyperion (disk)	\$22.95 now	\$19.44			
Name That Song (cass.)	\$14.95 now	\$12.64			
AVALON HILL GAME COMPANY					
B-1 Nuclear Bomber (cass.)	\$15.00 now	\$12.77			
Midway Campaign (cass.)	\$15.00 now	\$12.77			
No Atlantic Convoy Raider (cass.)	\$15.00 now	\$12.77			
Nukewar (cass.)	\$15.00 now	\$12.77			
Conflict 2500 (cass.)	\$15.00 now	\$12.77			
Planet Miners (cass.)	\$15.00 now	\$12.77			
Computer Acquire (cass.)	\$20.00 now	\$16.97			
Lords of Karma (cass.)	\$20.00 now	\$16.97			
AUTOMATED SIMULATIONS					
Introductory 3-Pack (disk)	\$49.95 now	\$39.97			
(Rescue, Morloc's, and Datestones)					
Rescue at Rigel (cass.)	\$29.95 now	\$25.47			
Temple of Apsahai (cass.)	\$39.95 now	\$33.97			
Hellfire Warrior (cass.)	\$39.95 now	\$33.97			
Starfleet Orion (cass.)	\$24.95 now	\$21.27			
Invasion Orion (cass.)	\$24.95 now	\$21.27			
Morloc's Tower (cass.)	\$19.95 now	\$16.97			
Datestones of Ryn (cass.)	\$19.95 now	\$16.97			
PERSONAL SOFTWARE					
VisiCalc (disk)	\$199.95 now	\$167.77			
Checker King (cass.)	\$19.95 now	\$16.97			
Gammon Gambler (cass.)	\$19.95 now	\$16.97			
MicroChess (cass.)	\$19.95 now	\$16.97			
Bridge Partner (cass.)	\$19.95 now	\$16.97			
Time Trek (cass.)	\$19.95 now	\$16.97			
UNITED SOFTWARE OF AMERICA					
KRAM (disk)	\$99.95 now	\$84.97			
Super KRAM (disk)	\$175.00 now	\$148.77			
Request (disk)	\$225.00 now	\$191.27			
Thinker (disk)	\$495.00 now	\$420.77			
Space Intruders (cass.)	\$19.95 now	\$16.97			
All MICRO-ED	10% Off List				
All Microcomputer Workshops	15% Off List				
BIG FIVE SOFTWARE					
Super Nova (cass.)	\$15.95 now	\$13.58			
Galaxy Invasion (cass.)	\$15.95 now	\$13.58			
Attack Force (cass.)	\$15.95 now	\$13.58			
Cosmic Fighter (cass.)	\$15.95 now	\$13.58			
Meteor Mission II (cass.)	\$15.95 now	\$13.58			
BRODERBUND SOFTWARE					
Galactic Trilogy (disk)	\$39.95 now	\$33.98			
Galactic Empire (cass.)	\$14.95 now	\$12.68			
Galactic Trader (cass.)	\$14.95 now	\$12.68			
Galactic Revolution (cass.)	\$14.95 now	\$12.68			
Tawala's Last Redoubt (cass.)	\$19.95 now	\$16.98			
DATASOFT					
Iago (disk)	\$24.95 now	\$21.18			
Football Classics (disk)	\$24.95 now	\$21.18			
Arcade-80 (disk)	\$24.95 now	\$21.18			
Iago (cass.)	\$19.95 now	\$16.98			
Football Classics (cass.)	\$19.95 now	\$16.98			
Arcade-80 (cass.)	\$19.95 now	\$16.98			
Sigmon (COLOR) (cass.)	\$29.95 now	\$25.38			
SECS (COLOR) (cass.)	\$29.95 now	\$25.38			
ACORN SOFTWARE					
Invaders From Space (disk)	\$20.95 now	\$17.78			
Duel-N-Droids (disk)	\$20.95 now	\$17.78			
Pinball (disk)	\$20.95 now	\$17.78			
Pigskin (disk)	\$20.95 now	\$17.78			
Quad (disk)	\$20.95 now	\$17.78			
Basketball (disk)	\$20.95 now	\$17.78			
Gammon Challenger (disk)	\$20.95 now	\$17.78			
Everest Explorer (disk)	\$20.95 now	\$17.78			
SuperScript (disk)	\$29.95 now	\$25.38			
System Savers (cass.)	\$14.95 now	\$12.68			
Invaders From Space (cass.)	\$14.95 now	\$12.68			
Duel-N-Droids (cass.)	\$14.95 now	\$12.68			
Pinball (cass.)	\$14.95 now	\$12.68			
Pigskin (cass.)	\$14.95 now	\$12.68			
Quad (cass.)	\$14.95 now	\$12.68			
Basketball (cass.)	\$14.95 now	\$12.68			
Gammon Challenger (cass.)	\$14.95 now	\$12.68			
Everest Explorer (cass.)	\$14.95 now	\$12.68			
All Adventure International	15% Off List				
All Automated Simulations	15% Off List				
All Avalon Hill	15% Off List				
All Hayden	15% Off List				
All Microsoft	15% Off List				
Apple					
Gorgon	\$39.95 now	\$33.99			
Word Star	\$375.00 now	\$289.00			
Mail Merge	\$125.00 now	\$106.19			
Super Sort	\$200.00 now	\$169.99			
VisiCalc 3.3	\$200.00 now	\$169.99			
Wurst of Huntington Computing	\$19.99				
Nibble Express	\$12.95 now	\$11.99			
Soft Porn Adventure	\$29.95 now	\$25.39			
Time Lord	\$29.95 now	\$25.39			
French Hangman	\$29.95 now	\$25.39			
Alicia-Sp. bilingual reader	\$29.95 now	\$25.39			
H&H Stock Trader	\$190.00 now	\$161.49			
Grow (CIA)	\$35.00 now	\$31.49			
Spelling Bee	15% Off List				
VersaCalc	\$100.00 now	\$84.99			
Hebrew	\$60.00 now	\$50.99			
All Serendipity	15% Off List				
Sneakers	15% Off List				
All Sybex Courses	15% Off List				
Win at the Races	\$39.95 now	\$33.99			
Disk Prep	\$25.00 now	\$21.19			
PLE Chip	\$60.00 now	\$50.99			

Special Birthday Sale
 For Two-Year-Old Computer Users
 To celebrate our daughter Melody's second birthday, we are offering one of the best games we've seen anywhere at a special price. DYNACOMP, maker of exceptionally fine computer software, makes a game called HODGE PODGE for children ranging from under two years old to primary grades. It runs on the Apple II* plus (Applesoft*) and needs 48K plus a disk drive. It lists for \$23.95 but if you say "Happy Birthday Melody" you can have it for \$18.99. We know you'll like it. We think it's fantastic.

Apple

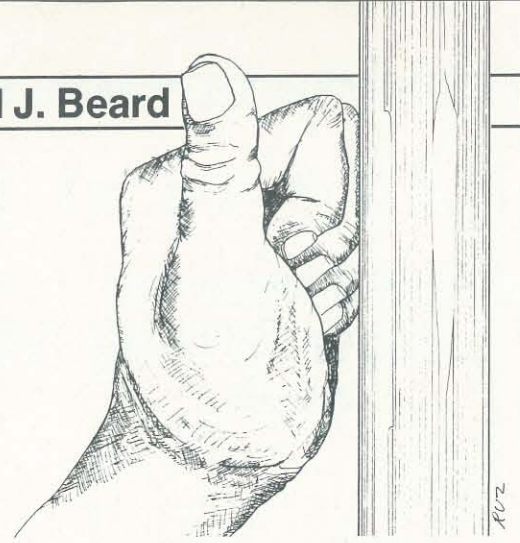
We maintain a huge inventory of Apple software and hardware. Call us toll free (outside Calif.) for the latest programs. We also stock a large supply of computer books. Visit us in person at our new 3300-square foot store at 1945 South Dairy in Corcoran, Calif.

Call Toll-Free 800-344-4111 (Outside California)

HUNTINGTON COMPUTING,
 Post Office Box 787
 Corcoran, California 93212
 Order by Phone 800-344-4111
 In California (209) 992-5411



We take MasterCard or VISA (include card # and expiration date). California residents add 6% tax. Include \$2.00 for postage. Foreign and hardware extra. Send for free catalog. Prices subject to change.



Computer Knock-Knock Jokes

LISP is a language which many people use to experiment with Artificial Intelligence. If you are interested in AI, or would just like to see LISP in action, or if you enjoy corny jokes, then read this article. The author provides a BASIC and a LISP version of the same program. Compare them, and see which intrigues you the most! — The Editor

Some time ago I heard rumors of a voice I/O terminal being demonstrated at a trade show. It seems that two computers equipped with the terminals were telling each other knock-knock jokes. At that time I was looking for simple problems to exercise a brand new, home-brew LISP interpreter, and the idea of the knock-knock jokes intrigued me. This article presents the program that resulted (Listing One), along with a very nearly equivalent program written in TRS-80 Model 1 Level 2 BASIC (Listing Two).

This is not a "LISP versus BASIC" article. The TRS-80 program was written to help explain the operation of the LISP functions by presenting a familiar equivalent. Taken by itself, it's a pretty poor piece of programming. The mere fact that one can write a short program similar to one in a radically different language says a great deal about the power and versatility of Microsoft's BASIC interpreter.

The first thing our program needs is a list of jokes. The function "FORGET" in Listing One means roughly, "set the symbol JOKES equal to an empty list." The NIL symbol is a sort of general negative in LISP. It means, among other things, "no," "none," "false," "empty," "the end," and "none of the above." Lines 10 through 150 of Listing Two accomplish much the same thing, except that we must be more specific about the type and number of items we want to store. Line 140 sets a pointer to the current "top of list." There are no dimension or type statements in LISP; however, the symbol used for our joke list must be assigned some specific value. There is no RUN or CLEAR function in LISP that will reset all variables to a known value, nor any distinction between symbols used for program elements and symbols used for data. To initialize our list, we would type "FORGET IT" in LISP, and "RUN" in BASIC.

Next, let's look at the function ASK in Listing One (lines 500-540 in Listing Two). The symbol INPUT in Listing One is a variable, not a function. ASK means roughly, "print out the value of A and read the operator's response. Set the symbol INPUT equal to this response, and return it to the calling function." A\$ and I\$ in Listing Two correspond directly to A and INPUT in Listing One. Line 520 clears I\$ so that the BASIC can detect "no input." The SETQ is necessary in Listing One because the READ function returns the value entered without assigning it to any symbol.

Let's get back to our joke list. The function KNOCK in Listing One and lines 200-290 of Listing Two are used to add new jokes to the joke list. The COND function is the LISP form of IF. KNOCK translates, "If there is an answer to the question 'who's there?' then SETQ JOKES, etc., else stop."

David J. Beard, RD#1, Box 648, Newmanstown, PA 17073.

The "etc." is a bit more complicated. Take a look at the first "knock-knock" in the sample run. The user types in "KNOCK KNOCK" to call the function KNOCK ("GOTO 100" in BASIC). "WHO'S THERE" is the argument passed to ASK and printed out by the computer. "ORANGE" is the value of INPUT at both places in the function. "ORANGE WHO?" is the value of the function APPEND. "ORANGE YOU ..." is the value of the second ASK. The second CONS adds the item "ORANGE" on to the beginning of the list "ORANGE YOU ..." and returns this new list to the first CONS, which adds it on to the beginning of the list JOKES. Finally, the function SETQ assigns this new joke list to the symbol JOKES. A literal English language equivalent for KNOCK would use at least four levels of subordinate clauses. The BASIC equivalent uses only one level of nesting, but uses five assignment statements to LISP's one. Assigning intermediate results to symbols simplifies the grammar enormously, both in the BASIC program and in my natural language explanation above. It also leads to a proliferation of symbols, which is one reason why the BASIC program uses numeric subscripts and pointers. It would be difficult to think of several hundred names for parts of a joke without getting downright silly. Notice that there is not a single numeric character in Listing One. This trade-off between complexity of grammar and the profusion of numerical pointers is a fundamental difference between LISP and the major procedural languages. I suspect that a choice between the two has more to do with personal taste than with the application involved.

The function TELL is more straightforward. In LISP one enters "TELL JOKES", in BASIC "GOTO 300". The DO in Listing One corresponds to the FOR-NEXT loop in Listing Two. DO will continue to evaluate the rest of the function until the final CDR A returns a NIL (until the remainder of A is an empty list). The COND in Listing One is directly equivalent to line 340 of Listing Two. "If there is no response to the prompt 'KNOCK! KNOCK!', stop." In LISP, the CAR of a list is its first element; the CDR is the rest of the list. Take a look at JOKES in Listing One. On a first pass, (CAAR A) is "ELIZA". (CDAR A) is "ELIZA DEAD ...". (CDRA) is the rest of the list, from "JOSE ..." on. (SETQ A (CDR A)) is equivalent to the "NEXT A" in BASIC. Notice that we are changing the value of A, not of JOKES, just as in BASIC we are changing the value of the loop-counter A, not the top-of-list pointer P.

There are many threaded languages that use named variables and procedures, but LISP offers an unusual degree of freedom from type declarations and resource allocation tasks. It is almost never necessary for the LISP programmer to know where a structure is stored or how much storage it's using. Program elements are lists exactly like data lists, and can be manipulated with the same primitives. If you're not turned off by the complex grammar (LISP = Lots of Infernal Stupid Parentheses), LISP offers a real alternative to the procedural languages with their cartesian arrays of data.

* KNOCK KNOCK!

Who's there?

Sam

Sam who?

Sam person who knocked last time.

Listing One

LISP version of

"Computer Knock-Knock Jokes"

```

*** JOKES ***
( ( ELIZA ELIZA DEAD PROGRAMMER. )
  ( JOSE JOSE CAN YOU SEE... )
  ( ORANGE ORANGE YOU READY YET? )
  ( BOO BOO HOO HOO! ) )

*** FORGET ***
( NFUN
  ( )
  SETQ JOKES NIL NIL )

*** KNOCK ***
( NFUN
  ( )
  COND
  ( ( ASK
    ( ( WHO'S THERE? ) )
    ( SETQ JOKES
      ( CONS
        ( CONS INPUT
          ( ASK
            ( APPEND
              ( WHO? )
              INPUT ) ) )
          JOKES ) ) ) )

*** TELL ***
( FUN
  ( A )
  DO
  ( ( COND
    ( ( ASK
      ( ( KNOCK! KNOCK! ) )
      ( ASK
        ( CAAR A ) )
      ( PRINTLIST
        ( CDAR A ) )
      ( PRINT )
      ( PRINT )
      ( SETQ A
        ( CDR A ) )
      ( CDR A ) ) ) ) )

*** ASK ***
( FUN
  ( A )
  ( PRINTLIST A )
  ( PRINT )
  ( SETQ INPUT
    ( READ ) )
  INPUT )

```

Sample run of LISP program

```

TINY LISP READY

JOKES
JOKES

TINY LISP READY

FORGET IT
JOKES

TINY LISP READY

KNOCK KNOCK
WHO'S THERE?
BOO
BOO WHO?
BOO HOO HOO!
JOKES

```

TINY LISP READY

```

KNOCK KNOCK
WHO'S THERE?
ORANGE
ORANGE WHO?
ORANGE YOU READY YET?
JOKES

```

TINY LISP READY

```

JOKES
((ORANGE ORANGE YOU READY YET?)
(BOO BOO HOO HOO!))

```

TINY LISP READY

```

TELL JOKES
KNOCK! KNOCK!
WHO'S THERE?
WHO'S THERE?
ORANGE
ORANGE WHO?
ORANGE YOU READY YET?

KNOCK! KNOCK!
WHO'S THERE?
BOO
BOO WHO?
BOO HOO HOO!

```

TINY LISP READY

Listing Two

BASIC version of

"Computer Knock-Knock Jokes"

```

10 REM ----- TELL KNOCK-KNOCK JOKES
20 REM ----- TRS-80 MOD 1 LEVEL II 16K
30 REM ----- 80 11 03 DJB
40 CLEAR 12000
100 REM ----- FORGET
110 DIM J$(255,2)
120 J$(1,1)=" "
130 J$(1,2)=" "
140 P=1
150 STOP
200 REM ----- KNOCK
210 A$="WHO'S THERE?"
220 GOSUB 500
230 IF I$="" THEN GOTO 290
240 J$(P,1)=I$
250 A$=I$+" WHO?"
260 GOSUB 500
270 J$(P,2)=I$
280 P=P+1
290 STOP
300 REM ----- TELL
310 FOR A=1 TO P-1
320 A$="KNOCK! KNOCK!"
330 GOSUB 500
340 IF I$="" THEN A=P:GOTO 390
350 A$=J$(A,1)
360 GOSUB 500
370 PRINT J$(A,2)
380 PRINT
390 NEXT A
500 REM ----- ASK
510 PRINT A$
520 I$=""
530 INPUT I$
540 RETURN

```

*KNOCK KNOCK!

Who's there?

Venice

Venice who?

Venice you going to open dis door?

* from Knock Knocks, The Most Ever, by William Cole. Dell, 1976.

Solutions continued from page 27

and proved in the 1600's that for any prime number P, $2 \uparrow (P-1)$ has a remainder of 1 when it is reduced by multiples of P. In our present problem, this means that $2 \uparrow (12)$ leaves 1 after harvesting groups of 13. Put another way, after 12 hours there is only 1 amoeba left. After 24 hours, there is again only 1 amoeba left.

How many 12's are there in 1000? $INT(1000/12)=83$. After $12*83=996$ hours there is again only 1 amoeba left. In the remaining 4 hours, the amoebas will grow to 16 and, after one more harvest, there will finally be 3 left.

Final Solution to Germ Harvest

```

10 INPUT X
20 REM * FIND REMAINDER OF X
  WHEN REDUCED BY 12 *
30 X=X-INT(X/12)*12
40 REM * LET THE AMOEBAS
  GROW X HOURS *
50 Y=2 ↑ X
60 REM *HARVEST THE EXCESS
  AMOEBAS *
70 Y=Y-INT(Y/13)*13
80 REM * ADJUST Y TO BE
  INTEGER *
90 Y=INT(Y+.5)
100 PRINT Y

```

Run this final solution with an input value of 1000 for X. Compare the time it takes to print the solution compared to that for the previous programs. This one runs almost as fast as when only one amoeba splits into two amoebas.

EDUCATIONAL SOFTWARE

TRS-80, COLOR COMPUTER, PET & APPLE II

ELEMENTARY	MATH
SCIENCE	BIOLOGY
GEOGRAPHY	HISTORY
ECONOMICS	ACCOUNTING
FOREIGN LANG.	BUSINESS ED.
GRAMMAR	FARM RECORDS
COIN INVENTORY	

Write for FREE Catalogue:
MICRO LEARNINGWARE, Box
 2134, N Mankato, MN 56001
 (507) 625-2205

VISA & MASTER CARD ACCEPTED

We pay 15% royalty for Educational Programs listed with us. TRS-80 is a registered trademark of TANDY CORP. PET is a trademark of COMMODORE BUS. MACHINES. APPLE is a trademark of APPLE COMPUTER CO.

For Photographers Only!



A Depth of Field Generator

From a hobbyist's viewpoint, there are two ways to consider a home computer. On one side, the micro can itself be considered a hobby; downing Klingons, solving Adventures, program writing and glitch tracing are all enjoyable and profitable ways to spend one's hobby time. Hardware enthusiasts with solder gun in hand also find relaxation (if you can call it that) by tinkering and improving on home computer systems.

On the other hand, a home computer can be used as a tool for the enjoyment of a separate and different hobby. While it is difficult to picture someone chucking happily to himself while he designs a new general ledger system, there are many fruitful applications whose enjoyment can be enhanced by a microcomputer. For example, hobbyists in fields such as astronomy and model airplanes (as in aerodynamics) find personal computers a valuable tool in the pursuit of their hobbies.

One field where a hobbyist can use a home computer is amateur photography. Commercial computers have long been

used in photography for lens design and table generation. While lens design is beyond the means and interests of most amateurs, table generation is applicable to an amateur and is easily performed on any personal computer.

In any photographic handbook you will find many general purpose photographic tables. Probably the most important tables for a beginning photographer are the depth of field tables. Depth of field is a fundamental concept of photography. Happily, depth of field tables can easily be generated by any home microcomputer. Before we look at how to generate customized depth of field tables, let's examine what depth of field is and why it is so important to photographers.

Depth of field is the range of acceptable focus in front of and behind the subject that the camera is focused on. If your subject is 15 feet away, your depth of field may be from 10 to 25 feet. Even though your camera is focused on 15 feet, objects between 10 feet away and 25 feet away will also be in focus.

Let's look at another case. Suppose you focus your camera on a person's face 2 feet away from you. Your depth of field may be such that objects from 1 foot 11 inches to 2 feet 1 inch away are in focus. Your subject's eyes will then appear crystal clear, but his nose and ears will be out of focus. It would be wise to chuck this particular photograph in the trashcan.

A good understanding of depth of field is essential to produce effective photographs. Creative use of depth of field can give your photographs a professional touch. How then, can you use depth of field to your advantage?

For any particular lens, depth of field is a function of two variables, focus distance and *f*/stop. Shutter speed must also be considered, as it directly affects the *f*/stop setting. When a camera is focused on a subject, focus distance is the distance between the camera and the subject. Unless either the camera or the subject is moving, this focus distance is constant when the photograph is being taken. Once the focus distance is determined, depth of field is directly defined by the *f*/stop setting and indirectly defined by the shutter speed. Since shutter speed and *f*/stop can be varied, depth of field can be varied. You may selectively alter the *f*/stop to create a desired depth of field. Let's look more at shutter speed and *f*/stop and see how they affect depth of field.

Shutter speed is a fairly simple concept; it is the amount of time that the shutter is open when a picture is snapped. A shutter speed of 1/60 second indicates the shutter will be open for 1/60 second.

f/stop, on the other hand, gives a lot of people trouble. Actually, *f*/stop is just as basic and simple as shutter speed. Think for a moment about what happens when you snap a picture. The shutter will open and allow light to pass through an opening to expose an image on a section of film. This opening, known as an aperture, can be of variable size. *f*/stop simply refers to the size of this opening. A large aperture will admit a greater quantity of light than a small aperture. The tricky part about *f*/stops is that *f*/stops are stated in inverse relation to the size of the aperture. For example, a high *f*/stop such as *f*/22 indicates a small aperture; a low *f*/stop such as *f*/2 or *f*/1.7 indicates a large aperture.

Now that we've introduced *f*/stops and shutter speeds, let's see how they can be used to obtain depth of field. To do this, we must understand what constitutes a proper exposure. In any situation, a certain total amount of light must be transmitted to enable a proper exposure. This quantity of light may be obtained by combining a large aperture with a fast shutter speed to admit a large portion of light for a short time. Conversely, a small aperture may be used with a slow shutter speed

to admit a small portion of light for a long (relative) period of time. In either case, the same total quantity of light is admitted. For instance, a camera setting of *f*/8 with 1/60 second may produce a proper exposure, as would several other combinations of *f*/stop and shutter speed. Since there can be many different settings that produce a proper exposure, how do you determine which is best?

This is where depth of field comes in. Generally speaking, the higher the *f*/stop setting, the wider the depth of field. Depth of field is dictated by the *f*/stop; since *f*/stop is affected by shutter speed, we can say that shutter speed indirectly affects depth of field.

Most beginning photographers will choose a shutter speed and then vary the *f*/stop to reach a correct exposure. By doing this, they lose direct control over depth of field. They should instead select an *f*/stop that will give them a desired depth of field, and then vary the shutter speed to provide a proper exposure.

Since most photographers want the widest depth of field possible, this policy is fairly easy to apply. When taking pictures, always use high *f*/stops combined with slow shutter speeds. This does not mean you are doomed to fuzzy pictures caused by slow shutter speeds; 1/60 second is slow and it is adequate for most situations. 1/125 second is still fairly slow yet it will freeze the spokes of a racing bicycle.

Let's imagine a scene where depth of field is used to compose a picture. You are at a white sandy beach dotted with several hilly dunes; warm waters are lapping gently at the shoreline. Fifteen feet away you see a pretty girl. These look like the ingredients for an attractive photograph, so enter Super-Photographer to the rescue! Grabbing your camera, you focus quickly on the girl, she flashes a bright smile and you're ready to shoot. Just then you notice an unsightly pile of seaweed about fifty feet away. From your vantage point, it will intrude into your picture. Without repositioning yourself or your model, you can use depth of field to salvage this shot.

Simply set your camera to obtain a narrow depth of field by using a high shutter speed and a low *f*/stop. Focus on the girl 15 feet away; your depth of field may now be such that everything from 12 to 20 feet away will be in focus. Now the objectional seaweed lump is just a formless blur against a soft, fuzzy background, providing you with an effective picture.

Depth of field can be effective in reverse, also. What if, instead of a messy seaweed pile, there were a majestic horse and rider emerging from behind a sand dune fifty feet away. Now the camera can be set to obtain a wide depth of field by using a high *f*/stop and a slow shutter speed. With a depth of field from 8 feet to infinity, you can snap your model with a sharply focused horse and rider in the background.

Since depth of field is such an important part of photography, all lenses have a depth of field scale painted on their tops. While this scale is often useful in applying your knowledge of depth of field, it doesn't provide you with the necessary detail to understand depth of field. (For example, a slide rule is useful in applying multiplication, but it doesn't help you understand multiplication.) It is for this reason that photographic handbooks contain depth of field tables. These tables can be studied and are a valuable tool in understanding the depth of field capabilities of different lenses.

The following BASIC program will allow you to generate your own depth of field tables. Of course, you may alter the parameters to personalize the tables for your own lenses; perhaps you may wish to examine the properties of a lens you wish to purchase. This program can be easily modified to run on any BASIC-equipped microcomputer. The equations used are found in any good photographic handbook. The user may

customize this program by changing three DATA statements near the end of the program: the table of *f*/stops, the table of lenses and the table of focusing distances. Note that lens sizes are stated in millimeters, and focusing distances are in feet (I try, but I still don't *think* metric); the program will perform the necessary conversions. Those of you desiring output on a screen may wish to change the PRINT statements, also. ■

```

10 PRINT
20 PRINT

30 REM *****
40 REM * BASIC PROGRAM TO DETERMINE NEAR DEPTH OF FIELD
50 REM * AND FAR DEPTH OF FIELD DEPENDING ON F/STOP,
60 REM * FOCUSING DISTANCE, AND FOCAL LENGTH OF LENS USED. *
70 REM * WRITTEN: 5/1/79 JOEY ROBICHAUX *
80 REM * VARIABLES: F1=F/STOPS, F2=FOCAL LENGTHS (METRIC) *
90 REM * F3=FOCAL LENGTHS (STANDARD), D1=FOCUSING DISTANCE *
100 REM * H2=HYPERFOCAL DISTANCE, I9=INFINITY *
110 REM *****

120 REM
130 DIM F1(13),F2(5),F3(5),D1(10)
140 REM
150 REM OBTAIN THE APERTURES (F/STOPS) AND STORE IN F1
160 FOR I1 = 1 TO 13
170 READ F1(I1)
180 NEXT I1
190 REM
200 REM OBTAIN THE FOCAL LENGTHS (IN F3)
210 REM AND CONVERT TO INCHES (IN F2)
220 FOR I1 = 1 TO 5
230 READ F3(I1)
240 LET F2(I1) = F3(I1)/25.4
250 NEXT I1
260 REM

270 REM OBTAIN THE FOCUSING DISTANCES AND STORE IN D1
280 FOR I1 = 1 TO 10
290 READ D1(I1)
300 NEXT I1
310 REM
320 REM PERFORM FOR THE NUMBER OF DIFFERENT LENSES
330 FOR I1 = 1 TO 5
335 REM
340 REM FOR EACH DIFFERENT LENS,
350 REM LET INFINITY = FOCAL LENGTH (IN MM) TIMES 10 FEET
360 LET I9 = F3(I1) * 10
370 PRINT "FOCAL LGTH","F/STOP","DIST","NEAR DOF","FAR DOF"
380 FOR I2 = 1 TO 13
390 REM
400 REM FIND HYPERFOCAL DISTANCE (H2)
410 LET H2 = F2(I1)*F2(I1)/F1(I2)*1000/12
420 FOR I3 = 1 TO 10
430 REM

440 REM FIND NEAR DEPTH OF FIELD (D2) AND FAR D.O.F. (D3)
450 LET D2 = H2*D1(I3)/(H2+D1(I3)-F2(I1)/12)+.041667
460 LET D3 = H2*D1(I3)/(H2-D1(I3)-F2(I1)/12)+.041667
470 REM
480 REM CONVERT D2 AND D3 TO FEET AND INCHES
490 REM USING INTEGER ARITHMETIC
500 LET I5 = INT(D2)
510 LET I6 = INT(D3)
520 LET I7 = INT((D2-I5)*12)
530 LET I8 = INT((D3-I6)*12)
540 REM
550 REM IF LESS THAN ZERO OR GT 10 TIMES FOCAL LENGTH,
560 REM MAKE IT EQUAL TO INFINITY (999*99")

570 IF I6 > 0 THEN 590
580 GOTO 600
590 IF I6 < I9 THEN 620
600 LET I6 = 999
610 LET I8 = 99

620 PRINT F3(I1),F1(I2),D1(I3),I5;" ";I7;"",I6;" ";I8;""
630 NEXT I3
640 NEXT I2
650 PRINT
660 NEXT I1
670 REM
680 REM TABLE OF F/STOPS

690 DATA 1.4,1.7,2.0,2.8,3.5,4.0,5.6,6.3,8.0,11.0,16.0,22.0,32.0
700 REM
710 REM TABLE OF LENSES (IN MM)
720 DATA 28,35,50,80,135
730 REM
740 REM TABLE OF FOCUSING (OR SUBJECT) DISTANCES
750 DATA 2,5,8,10,13,15,20,25,40,70
760 END

```

Fibonacci Nim



Nim is a classic computer game. This version allows several people to play at once, making it excellent for get-togethers, demonstrations, hands-on learning sites, etc. Much could be said about the mathematical theory and strategy one learns from Nim, but mainly . . . it's fun!

Fibonacci Nim was invented by Robert E. Gaskell of Oregon State University and is discussed in Martin Gardner's *Mathematical Circus* ("Fibonacci and Lucas Numbers," pages 164-167). The game involves two players and a pile of objects. The players take turns removing objects from the pile and the person who removes the last object wins the game. Each player must take at least one of the objects on each turn but cannot take more than twice the number of objects just taken by the opponent; for example, if one player takes five objects, then the next player cannot take more than ten. The first player cannot take the whole pile but can take any smaller (positive integer) number of objects.

The program allows the computer to play against many people simultaneously. Each person gets to name the type of objects (pizzas, diamonds, French fries, etc.) and the computer remembers the type of object associated with each person. People need not play in turn; when a person has decided on her or his move, the person tells the computer and the computer reviews the game situation and accepts the move (if the move is acceptable). Each person is played three games. The program allows a few more players than it was initially told of, and as players finish their round of three games, storage space for new players is made available.

Suppose the program is to be demonstrated all day in a shopping mall. We would anticipate that some players will wander off without finishing their three games. The demonstrator can type "LIST NAMES" in response to "Who's next?" and get a list of the players - this is useful if someone has forgotten how his name was entered. The program then asks if any of the names should be deleted; the names of missing players can be removed to make room for new players. If a new player has the same name as an old player, the program asks for an additional name to distinguish the two players.

In the first game, the program selects the pile size and the human chooses who goes first; thereafter, the winner selects the pile size and the loser picks who starts. The program is satisfied with one win out of the three games; most people can

beat the program one or two times but almost no one can beat the program all three times.

The program given here is for a Zenith/Heath H89 with MBASIC (c) but is easily modified to TRS-80 Level Two BASIC; the apostrophes act like REM's and the colons separate multiple commands. All of the displays are designed for 16 lines and 64 columns (such as found on the Model I and III TRS-80's); the initial display uses an animated dash (produced by line 2160) to allow viewer time to glance at the credits before seeing a sixteen-line frame of rules.

The 3000-3999 block explains H89-related material. The "PRINT CS\$" (defined in line 2010) clears the screen; the screen never really needs clearing so you can delete or substitute as you like. The FCNLS() can be deleted if all input is in capital letters; it maps lower-case input into upper case. Line 2030 performs the function that many BASIC's handle by "RANDOM" - seeding the random number generator. H89 MBASIC uses RND(1) where the TRS-80's would use RND(0).

I guess that Michael J. Whinihan, who described the optimal strategy in the December 1963 issue of *The Fibonacci Quarterly*, could beat the program three times in a row, provided that he was careful and knew some large Fibonacci numbers. Can you beat it three times in a row?

Welcome to the Game of FIBONACCI NIM!

The game begins with a pile of objects. The play alternates between the two players and consists of removing objects from the pile. The first player cannot take the entire pile and at least one object must be taken on each play. A player may not take more than twice the number of objects that his or her opponent took on the last play. For example, if one player picks up three objects, the next player may not take more than six objects. The player who takes the last object wins!

When ready for more information, press the 'ENTER' button.

We will play three games. For the first game, I'll choose the pile size and you choose who goes first or second. Thereafter, the winner chooses the pile size and the loser chooses who starts.

Just to make it more fun, I guarantee that I won't lose all three games.

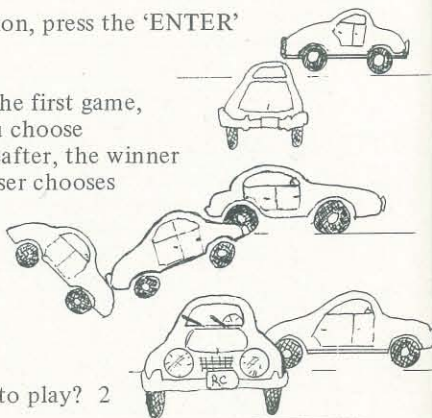
Are you ready to play? yes

How many people might want to play? 2

Ready to start? yup

The pile of objects we are playing with can be a pile of anything. For example, sticks, books, apples, diamonds, toys . . . What kind of objects do you want in your pile? green cars

If you want green cars, Lisa, you can have them!



Ok, Lisa, since this is the first game, I'll choose the pile size and you choose who starts. Here we go! I choose a pile size of 30. Do you want to start? yes

It's your turn, Lisa, you have 30 green cars and you may take up to 29 green cars. How many green cars do you want to take? 9

Hmm, that was an interesting move, Lisa. Let's see, there are 21 green cars and I can take up to 18 green cars. This is going to be tough! I'm going to pick up just 1 this time. That'll leave you with 20 green cars, and the possibility of picking up at most 2 green cars.

Who's next? Johnny

Great! A new player! Welcome Johnny! Do you know how to play? yes The pile of objects we are playing with can be a pile of anything. For example, sticks, books, apples, diamonds, toys . . . What kind of objects do you want in your file? French fries

If you want French fries, Johnny, you can have them! Ok, Johnny, since this is the first game, I'll choose the pile size and you choose who starts. Here we go! I choose a pile size of 41. Do you want to start? nope

Nice of you to let me start, Johnny. Let's see, there are 41 French fries and I can take up to 40 French fries. Let me think about this . . . I'm going to pick up 2 French fries. That'll leave you with 39 French fries, and the possibility of picking up at most 4 French fries.

Who's next? Lisa

As I remember, Lisa, we are in game 1 and it's your turn with 20 green cars to work with. Is that correct? Yes

It's your turn, Lisa, you have 20 green cars and you may take up to 2 green cars. How many green cars do you want to take? 2

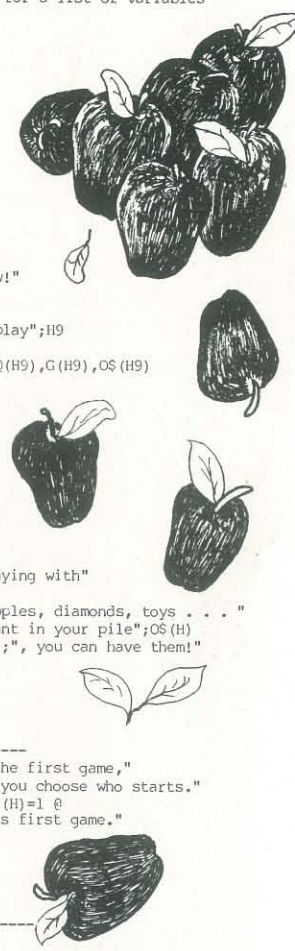
That was a good play, Lisa! Now, it's my turn. Let's see, since you took 2 green cars, I can't take more than 4 green cars. I'm going to pick up just 1 this time. That'll leave you with 17 green cars, and the possibility of picking up at most 2 green cars.

Who's next? Connie

```

1 '          *** FIBONACCI NIM ***          @
2 ' written by Mary Ann Gohlke             @
3 ' in the Spring of 1980 while she was a  @
4 ' senior at the University of Houston Victoria Campus @
5 ' Original program, for a 16 computer-k level II TRS-80, @
6 ' submitted to RECREATIONAL COMPUTING.
7
8 CLEAR 500
9 GOSUB 2000 ' modifications made by Furman Smith at UHVC @
10 ' for the Zenith/Heath All-in-One
11
12 GOSUB 9000 ' RULES
13
14 ----- INITIALIZATION -----
15 ' see end of the program for a list of variables
16
17 F9=20
18 DIM F(F9)
19 B1=0
20 B2=1
21 FOR I=1 TO F9
22 F(I) = B1+B2
23 B1=B2
24 B2=F(I)
25 NEXT I
26
27 INPUT "Are you ready to play";AS
28 IF AS="" THEN AS="Y"
29 AS=FNLS(AS) : IF AS="Y" GOTO 90
30 INPUT "Does anybody want to play";AS
31 AS=FNLS(AS) : IF AS="Y" GOTO 90
32 PRINT "Sorry no one wants to play,"
33 PRINT "It's really a fun game."
34 PRINT "Maybe some other time. Bye now!"
35 STOP
36
37 INPUT "How many people might want to play";H9
38 H9=H9+2
39 DIM S(H9),W(H9),NS(H9),F1(H9),T(H9),Q(H9),G(H9),OS(H9)
40 FOR I=1 TO H9
41 S(I)=1 : W(I)=0 : F1(I)=0
42 NEXT I
43 H=1 : I9=1
44
45 INPUT "Ready to start";AS
46 AS=FNLS(AS) : IF AS="Y" GOTO 140
47 PRINT "I'll wait . . ." : GOTO 125
48
49 INPUT "What's your name";AS
50 NS(H)=AS
51 PRINT "The pile of objects we are playing with"
52 PRINT "can be a pile of anything."
53 PRINT "For example, sticks, books, apples, diamonds, toys . . ."
54 INPUT "What kind of objects do you want in your pile";OS(H)
55 PRINT "If you want ";OS(H);", ";NS(H);", you can have them!"
56 PRINT :PRINT
57
58 ON S(H) GOTO 200,300,400,500
59
60 ----- FIRST GAME -----
61 PRINT "Ok, ";NS(H);", since this is the first game,"
62 PRINT "I'll choose the pile size and you choose who starts."
63 B2=INT(1+10*FND(1)) : IF B2>6 THEN F1(H)=1 @
64 : PRINT "I'll even go easy on you this first game."
65
66 PRINT "Here we go!"
67 GOSUB 600
68 GOTO 1000
69
70 ----- SECOND GAME -----
71 IF W(H)>0 GOTO 355
72 PRINT
73 PRINT "Since I won the last game,"
74 PRINT "I'll choose the pile size and you choose who starts."
75 B2=INT(1+10*FND(1)) : IF B2>5 THEN F1(H)=1 @
76 : PRINT "I'm going to give you a chance to win -- BE CAREFUL !"
77 IF B2>5 GOTO 340
78 PRINT "Unless you want to choose the pile size - "
79 PRINT "guess we could bend the rules a little."
80 PRINT "Course, then I'd get to choose who starts."
81 INPUT "Do you want to choose the pile size";AS
82 AS=FNLS(AS) : IF AS="Y" GOTO 397
83 GOSUB 600 : GOTO 1000
84
85 INPUT "Are you ready for game two";AS
86 AS=FNLS(AS) : IF AS="Y" GOTO 387
87 INPUT "Aren't you going to give me a chance to win";AS
88 AS=FNLS(AS)
89 IF AS="Y" THEN PRINT "Thanks ";NS(H);"! Here we go!" :GOTO 390
90 PRINT "Aw, come on ";NS(H);", since we agreed to 3 games,"
91 PRINT "it is only fair to continue."
92 PRINT "Here we go with game two!"
93 PRINT "Let's see, since you won the last game,"
94 PRINT "you get to choose the pile size and I pick who starts."
95 GOSUB 655 : GOTO 1000
96
97 ----- THIRD GAME -----
98 IF W(H)>10 GOTO 420
99 F1(H)=1
100 PRINT "Don't give up ";NS(H);","
101 PRINT "I'll go easy on you this time."
102 PRINT "Here we go with game three!"
103 IF W(H)>=10 GOTO 390 ELSE GOTO 395
104

```



Roman Numeral Conversum Programum

*in asic-Bay for your
omputer-Cay*

Up until the 1500's, when Arabic notation finally came into wide usage, Roman numerals were the most widespread and popular method of counting. Today the roman numeral is still used for copyright dates in movies and books; in certain documents; on some statues; and, of course, are a necessity in the study of medieval scrolls. None of which is exceptionally impressive. In fact, one wonders how the Romans managed to conquer the known world with such a cumbersome mathematical system. Hey! They didn't even have a "zero." 'S truth.

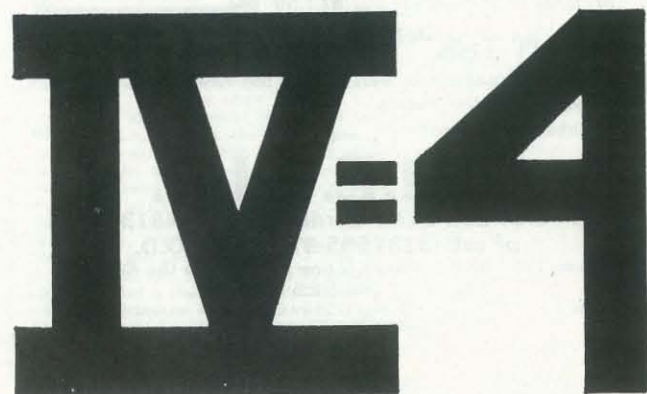
But, think of this. What if the Romans had possessed personal computers to easily convert their unwieldy numerical system to more usable notation? Right! We'd probably all be speaking Latin today.

So. Here's a computer program to convert Roman numerals to Arabic (that's like 1, 2, 3) and vice versa. Give 'er a try for fun. I think you're a little late on conquering the known world, tho'.

The program's actually less complex than it might look at first glance. It's written in Computerware 8.5 BASIC for my Smoke Signal Chieftain but should work in other BASICs with minimal change.

One note. In their notation, the Romans made use of the "vinculum" (yessir, a real Latin word!). The vinculum is a bar over part of the numerals and indicates that *only* the symbols under the bar are to be multiplied by 1000. The symbols not under the vinculum are added to this total. By this device, they could indicate higher numbers with fewer letters. It's still a clumsy system. Anyway, I've used parentheses instead. Probably what the Romans would've done if they had thought of it!

Once you get the program up and running, try converting a number like, for example, 888363. The answer will graphically demonstrate why the fur-clad barbarians from the north on their sturdy little ponies were able to finally grind the Roman civilization under.



BASIC
#RUN

ROMAN NUMERAL CONVERSION PROGRAM

Convert FROM or TO Roman Numerals (1 or 2)? 1

If a bar is over any of the number enter '1', a '0' if not? 0
What Roman Numeral do you wish converted? XVIII

XVIII = 18

Want another (1= 'yes')? 1

ROMAN NUMERAL CONVERSION PROGRAM

Convert FROM or TO Roman Numerals (1 or 2)? 1

If a bar is over any of the number enter '1', a '0' if not? 0
What Roman Numeral do you wish converted? CDLXXVIII

CDLXXVIII = 478

Want another (1= 'yes')? 1

ROMAN NUMERAL CONVERSION PROGRAM

Convert FROM or TO Roman Numerals (1 or 2)? 1

If a bar is over any of the number enter '1', a '0' if not? 1
Enter ONLY the letters UNDER the 'vinculum'?

DCCCLXXXVIII

Enter the rest of the letters (those not under the bar)?

CCCLXIII

(DCCCLXXXVIII)CCCLXIII = 888363

Want another (1= 'yes')? 1

ROMAN NUMERAL CONVERSION PROGRAM

Convert FROM or TO Roman Numerals (1 or 2)? 2

What is the Arabic Number to be converted to ROMAN? 1981

1981 = MCMLXXXI

Want another (1= 'yes')? 1

ROMAN NUMERAL CONVERSION PROGRAM

Convert FROM or TO Roman Numerals (1 or 2)? 2

What is the Arabic Number to be converted to ROMAN?
888363

888363 = (DCCCLXXXVIII)CCCLXIII

Remember number in para. is equiv. to 'vinculum'

```
0010 REM :: ROMAN NUMERAL CONVERSION ::
0020 REM :: by Ralph Roberts MCMLXXXI ::
0030 FOR X=1 TO 9:READ I$(X):NEXT X:FOR X=1 TO 9:READ XS(X):NEXT X
0040 FOR X=1 TO 9:READ D$(X):NEXT X:READ M$:LINE=90:P.:P.
0070 PRINT TAB(25);"ROMAN NUMERAL CONVERSION PROGRAM":P.
0090 INPUT "Convert FROM or TO Roman Numerals (1 or 2)":I
0100 IF I<1 THEN 90
0110 IF I>2 THEN 90
0120 PRINT :PRINT:IF I=2 GOTO 520
0130 REM :: main routine for converting FROM ::
0140 INPUT "If a bar is over any of the number enter '1', a '0' if
not",A1
0150 IF A1=1 GOTO 470
0160 INPUT "What Roman Numeral do you wish converted",B$
0170 LET A=LEN(B$):FOR X=1 TO A:LET C$=MID$(B$,X,1)
0190 IF C$="M" LET T=T+1000:GOTO 290
0200 IF C$="D" LET T=T+500
0210 IF C$="C" GOSUB 350
0220 IF C$="L" LET T=T+100
0230 IF C$="X" LET T=T+10
0240 IF C$="I" GOSUB 390
0250 IF C$="V" LET T=T+5
0260 IF C$="X" LET T=T+10
0270 IF C$="I" GOSUB 430
0280 IF C$="V" LET T=T+5
0290 NEXT X
0300 IF A1=1 GOTO 490
0310 PRINT :PRINT:IF A2=1 PRINT(":"A$;")";
0330 PRINT B$;"=";"":T;GOTO 810
0350 LET F$=MID$(B$,X+1,1)
0355 REM :: 3 subroutines for '4' & '9' anomalies ::
0360 IF F$="M" LET T=T+900:X=X+1:GOTO 290
0370 IF F$="D" LET T=T+400:X=X+1:GOTO 290
```

```
0380 RETURN
0390 LET F$=MID$(B$,X+1,1)
0400 IF F$="C" LET T=T+90:X=X+1:GOTO 290
0410 IF F$="L" LET T=T+40:X=X+1:GOTO 290
0420 RETURN
0430 LET F$=MID$(B$,X+1,1)
0440 IF F$="X" LET T=T+9:X=X+1:GOTO 290
0450 IF F$="V" LET T=T+4:X=X+1:GOTO 290
0460 RETURN
0470 REM :: 'vinculum' subroutine ::
0480 INPUT "Enter ONLY the letters UNDER the 'vinculum'",A$:B$=A$:GOTO
170
0490 LET A1=0:T=T*1000:LET A2=1
0510 INPUT "Enter the rest of the letters (those not under the bar)",
B$:GOTO 170
0520 REM :: Main Routine for Converting ARABIC to ROMAN ::
0550 INPUT "What is the Arabic Number to be converted to ROMAN",B$
0555 LET A=LEN(B$):DIM V$(A)
0560 FOR X=A TO 1 STEP -1
0570 LET C$=MID$(B$,X,1)
0580 IF C$="0" GOTO 730
0590 IF X<A GOTO 620
0600 LET V$(X)=I$(VAL(C$)):GOTO 730
0620 IF X<A-1 GOTO 650
0630 LET V$(X)=X$(VAL(C$)):GOTO 730
0650 IF X<A-2 GOTO 680
0660 LET V$(X)=D$(VAL(C$)):GOTO 730
0680 REM :: trip for 'vinculum' ::
0690 IF VAL(B$)>2999 GOTO 830
0700 FOR Y=1 TO VAL(C$)
0710 LET V$(X)=V$(X)+M$:NEXT Y
0730 NEXT X:P.:P.:P.
0750 IF A2=1 PRINT A$;"=";"":GOTO 770
0760 PRINT B$;"=";"":
0770 IF A2=1 PRINT "(";
0780 FOR X=1 TO A:P.V$(X):NEXT X
0790 IF A2=1 PRINT ")"";Z$
0800 IF A2=1 PRINT:P."Remember number in para. is equiv. to 'vinculum'"
0810 PRINT :P.:INPUT"Want another (1= 'yes')";I:IF I=1 RUN
0820 END
0830 REM :: subroutine for Arabic Numbers 3000 & over ::
0835 REM :: sets up 'vinculum' ::
0840 LET A2=1:FOR Z=1 TO A:LET Z$=Z$+V$(Z)
0850 NEXT Z:A$=B$:A2=1
0860 LET A3=INT(VAL(B$)/1000):LET B$=STR$(A3)
0870 FOR Z=1 TO A:LET V$(Z)="":NEXT Z
0880 LET A=LEN(B$):GOTO 560
0900 DATA I,II,III,IV,V,VI,VII,VIII,IX
0910 DATA X,XX,XXX,XL,L,LL,LXX,LXXX,XC
0920 DATA C,CC,CCC,CD,D,DC,DCC,DCCC,CM,M
```

**ATTENTION
EDUCATORS**

COMPUTER PROGRAMS
GRADES 5-13

THE VOCABULARY GAME- With the motivation of a sports format, students have fun while improving their vocabulary.

THE FRENCH GAME — A vocabulary game for beginning French students that makes learning French exciting.

C.A.I. CHEMISTRY*—15 programs on the main course topics. A new 48K HI-RES version is now available.

C.A.I. BIOLOGY — 15 programs on 15 major topics. Keeps track of student progress. A new HI-RES version is available.

PHYSICS — Programs for the high school student.

THE APPLE GRADE BOOK- The complete grade book for all teachers.

All Available for the APPLE II

FOR INFORMATION WRITE
J & S SOFTWARE
140 Reid Ave.
Port Washington,
N.Y. 11050

*CHEMISTRY is also available for the TRS-80

THE COMPUTING TEACHER

The Computing Teacher is a journal for educators interested in teaching using computers and teaching about computers. It is aimed mainly at the precollege level, elementary and secondary school. Each issue carries material of interest to elementary school teachers, to secondary school teachers, and to teachers of teachers.

The Computing Teacher is published by the International Council for Computers in Education, which is a non-profit corporation dedicated to the increased and improved use of computers in education. The journal also carries material on use of calculators.

The Computing Teacher will publish nine issues during the academic year 1981 - 1982. The publication is now in its ninth year.

Subscription Rates:	United States	Foreign
	\$14.50 (9 issues)	\$20.00
	\$27.50 (18 issues)	\$38.00
	\$38.00 (27 issues)	\$59.00

The Computing Teacher
% Computing Center
Eastern Oregon State College
La Grande, Oregon 97850

Name _____
Address _____
City _____
State _____ Zip _____

A Game Machine



One use for Radio Shack's pocket computer is to entertain and to educate. The pocket computer is easy to carry and can be used on a train or while waiting in a restaurant. It can be a very portable game machine. I have used mine several times with my nine year old when situations started to become awkward.

This use for the pocket computer depends upon one's ability to think up simple games. My first idea was multiplication practice. The computer would ask what range of numbers one wished to multiply, then it would generate random pairs of numbers and ask their product. A correct response would cause the computer to give a favorable reply. An incorrect response would cause the question to be asked again.

The multiplication practice program required a random number generator. This could also be used for a "guess the mystery number" game. To adapt the game to a wide spectrum of abilities, the program first asks for the range of possible choices for the mystery number. My young son enjoys watching and seeing me struggle trying to find the mystery number between one and a million. With each guess of the mystery number, the pocket computer responds with either "too high" or "too low" and counts the number of guesses. It requires quite a bit of concentration to find a mystery number between one and a million in twenty guesses. Trying to find the mystery number between one and a hundred is an easier game for children.

My ideas for games are limited, so my next idea was a game to find a pair of mystery numbers. To make this game sound more interesting, the player tries to drop a bomb on a mystery submarine. A mystery submarine starts somewhere

on a grid whose horizontal and vertical coordinates each range from -1000 to +1000. The player guesses the horizontal and vertical coordinates for the bomb, and if the bomb falls within 20 units (or yards) of the mystery submarine, the submarine is destroyed. (If someone objects to bombing submarines, we could as easily dig for buried treasure.) After each guess of the coordinates, the pocket computer tells the player by how much he missed. The computer counts the number of bombs dropped, and the object of the game is to destroy the mystery submarine with the minimum number of bombs. As an additional feature, I put in a few lines of program which cause the submarine to move ten units in a random direction before each bomb. Unfortunately, the result is a fairly difficult game and I still haven't found a suitable strategy for playing the game.

My last idea was more of a simulation than a game. I call it the Jumping Frog Contest. We have three frogs in a race from zero to one hundred. The output of the random number generator is modified to cause each frog to jump anywhere from ten units forward to four units backward. Before the race the players can try to guess which frog will win. After that it just consists of watching the race as reported by the display of the computer.

The subroutine central to all of the games is a random number generator. I use a crude and unsophisticated random number generator based on taking the fractional part of $(\pi + s)^5$ where s is the seed. This random number between zero and one is then multiplied by L to obtain a random integer between one and L . This subroutine is as follows:

```
500: N = (PI + S) ↑ 5
510: S = N - INT(N)
```

```
520: R = INT(L * S + 1)
530: RETURN
```

It appears to give a satisfactory set of "random" integers.

The first part of the program in the pocket computer is the Multiplication Practice. Most of the program consists simply of communicating with the player and giving the proper response. Lines 190, 191, and 192 give three different responses for a correct answer. The program is as follows:

```
10: S = .123456
20: PAUSE "HELLO THERE"
30: INPUT "WHAT IS YOUR NAME? ", A$
40: PAUSE "I WOULD LIKE TO HELP"
50: PAUSE "YOU PRACTICE"
60: PAUSE "MULTIPLICATION "; A$
70: PAUSE "WHAT IS THE LARGEST"
80: PAUSE "NUMBER YOU WISH TO"
90: INPUT "MULTIPLY? "; L
100: GOSUB 500
110: X = R
120: GOSUB 500
130: Y = R
140: PAUSE "FIND "; X ; " "; Y
145: PAUSE "FIND "; X ; " "; Y
150: INPUT P
160: IF P=X*Y GOTO 190+INT(3*S)
170: PAUSE "TRY AGAIN "; A$
180: GOTO 140
190: PAUSE "GOOD WORK "; A$ ; GOTO 100
191: BEEP 2 : PAUSE "THAT IS RIGHT "; A$ : GOTO 100
192: PAUSE "CORRECT ANSWER "; A$
200: GOTO 100
```

The next section of the program is the Jumping Frog Contest. The frogs are 1, 2, and 3; and their positions are A(1), A(2), and A(3). The program is as follows:

```
300: PRINT "FROG JUMPING CONTEST"
310: PAUSE "WHICH FROG WILL"
320: PAUSE "WIN THIS TIME?"
330: FOR I = 1 TO 3
340: A(I) = 0
350: NEXT I
360: FOR I = 1 TO 3
370: L = 15
380: GOSUB 500
390: J = R - 5
400: A(I) = A(I) + J
410: PAUSE "FROG "; I ; " IS AT "; A(I)
420: IF A(I) >= 100 THEN 450
430: NEXT I
440: GOTO 360
450: PRINT "THE WINNER IS FROG "; I
460: GOTO 300
```

The section of program after the random number generator is the Number Guessing Game. Line 706 allows the player to decide if he wants to guess numbers between one and one hundred or between one and one million. The player inputs his guess on line 720. The program goes as follows:

```
700: PRINT "NUMBER GUESSING GAME"
702: PAUSE "WHAT IS THE LARGEST POSSIBLE"
704: PAUSE "MYSTERY NUMBER YOU WISH"
706: INPUT "TO GUESS? "; L
710: GOSUB 500
715: G = 0
720: INPUT "GUESS MYSTERY NUMBER "; Y
725: G = G + 1
730: IF Y=R GOTO 760
740: IF Y<R GOTO 770
750: IF Y>R GOTO 780
760: PAUSE "YOU FOUND IT"
765: PRINT "NUMBER OF GUESSES="; G
767: GOTO 700
770: PAUSE "YOU ARE TOO LOW"; GOTO 720
780: PAUSE "YOU ARE TOO HIGH"; GOTO 720
```

The last section of program in my pocket computer is the Submarine Game. Here the object of the game is to destroy the hidden submarine with the minimum number of bombs. The submarine is destroyed when a bomb falls within 20 yards of the submarine. The coordinates of the submarine are U and V. On lines 810 and 830 they obtain their initial random values somewhere between -1000 and +1000. The submarine is moving and the coordinates U and V are changed slightly on lines 835 and 837. On line 850, the distance between the bomb and the submarine is measured and this distance is reported on line 870. This section of program is as follows:

```
800: L=2000 : PRINT "SUBMARINE GAME"
801: PAUSE "GIVE COORDINATES X,Y"
802: PAUSE "FOR BOMBS TO DESTROY"
803: PAUSE "MYSTERY SUBMARINE"
804: PAUSE "- 1000 < X < 1000"
805: PAUSE "- 1000 < Y < 1000"
806: PAUSE "SUBMARINE IS MOVING"
808: GOSUB 500
810: U=R-1000 : M=1
```

```
820: GOSUB 500
830: V=R-1000
831: L=360
832: GOSUB 500
833: T=R
835: U=U+10*COS(T)
837: V=V+10*SIN(T)
840: INPUT "GIVE COORDINATES X,Y? "; X,Y
850: D=INT(SQRT((X-U)*(X-U)+(Y-V)*(Y-V)))
860: IF D<20 PRINT "YOU HIT IT" : GOTO 895
870: PRINT "YOU MISSED BY "; D ; " YARDS"
880: PAUSE "BOMBS DROPPED="; M
890: M=M+1 : GOTO 835
895: PRINT "BOMBS DROPPED="; M ; GOTO 800
```

These programs virtually fill the available memory in the pocket computer. There are 39 steps of available memory remaining.

The seed on line 10 can be changed to make the random numbers unpredictable. This is usually not necessary since most children will not question the randomness of the numbers. The fascination of playing games with a tiny computer seems to hold the interest of children of all ages.

PET® and TRS-80® Programs

Present this ad from R.C. and receive \$2 off your purchase price. Valid at your local dealer or when ordered direct.

- **PROFESSIONAL TOOLS**
 - Business Researcher.....(16k) \$50
 - RNAV3 Navigator.....(16k) \$30 (8k) \$25
 - Education Pack (High School).....\$15
- **DISK BOWLING SYSTEM (PET/CBM)**
 - Leaguebowl-24.....(Disk 32k) \$145
 - Archivebowl.....(for above) \$40
 - Leaguebowl-12.....(Cass. 16k) \$40
 - Tournamentbowl.....(Cass. 8k) \$30
- **HOME & OFFICE**
 - Deluxe Address (16k)\$40
 - Home Address.....\$25
 - Grocery Mart.....\$15
 - Inventory.....\$20
 - Shopper.....\$20
 - Dinner's On!.....\$15
- **GAMES & SIMULATIONS**
 - Mansion!.....\$15
 - Museum!.....\$15
 - Pentagon!.....\$15
 - Fur Trapper.....\$15
 - High Seas.....\$15

Send for free catalog!

HARRY H. BRILEY
P.O. Box 2913
Livermore, CA 94550
(415) 455-9139

Dealers: Letterhead inquiries invited. Photocopies of this ad are NOT valid coupons. One coupon per purchase. This coupon may be redeemed for face value plus 15¢ for handling if it was received from customer upon purchase of one of the above programs. Offer void where restricted by law.

PET: Commodore Trademark; TRS-80: Tandy Corp. Trademark

CLASS OF '82... BOOT UP!

ASSISTANT PRINCIPAL - Whether you're too small for a big computer or if you just want to bring your administrative duties under your control, the "Assistant Principal" is just what you need. It prints school rosters, allows you to input teacher grades, prepare report cards, and maintain student master records. You should have no more trouble with student scheduling, grade averaging or recording of student grades. For Apple II and Apple II Plus, two disk drives, 48K Memory and an 80 column printer. Item M-3839-11. Price \$500.00.

ROLL CALL - Let this program keep track of that all important attendance data. Whether you need information on an individual's tardy, absence history or need to prepare attendance reports for the board or your state funding agency, this complete system will make the task easy work. The system will create its own data base or work off the Assistant Principal's files. For Apple II or Apple II Plus, two disk drives, 48K Memory, and an 80 column printer. Item M-52. Price \$250.00.

SYSTEMS ANALYST PLUS - Whether you're a novice computer student, teacher or principal, you will find this program most helpful. Even the advanced programmer should find these seventeen teaching sessions on programming useful for brushing up on the language and commands. For Apple II or Apple II Plus with Applesoft, one disk drive, and 32K Memory. Item M-40. Price \$69.95.

Call our toll free number for a free catalog on professional, business, accounting and educational programs.

(800) 854-0561
In Ca. (800) 432-7257
Ext. 802 or write:

MONUMENT COMPUTER SERVICE
Village Data Center
P.O. Box 603
Joshua Tree, California 92252

Commodore Links Up

Commodore Business Machines and the University of Waterloo, Ontario have teamed up to produce a product package of a sort that's long overdue: a consistent, well-integrated set of software that makes true partners of a micro and a large time-sharing system. They've afflicted the package with the awkward title of "Micro-Mainframe," but — assuming it can be delivered as advertised — it has far-reaching implications.

The U. of Waterloo has long been a leader in the design of time-sharing software for big machines; their WATFOR and WATFIV FORTRAN systems are used at many schools. They provided the specifications for the system, while Commodore provided the hardware and Waterloo Computing Systems Limited is developing the software. The hardware is based on the CBM 8032 business micro with 96K of storage and the usual CBM peripherals.

The software package is remarkable for its scope; it includes interpreters for four different programming languages, each in two versions: one for the micro and one for the mainframe. The languages are a BASIC (not, apparently, compatible with PET BASIC), a full standard Pascal, a FORTRAN dialect "designed for teaching purposes," and a "complete and faithful implementation" of APL (hip, hip, hurrah!). A relocating macro assembler and a linkage editor, major products themselves, are mentioned as an afterthought.

This cornucopia of software (full APL, oh boy!) is claimed to be available in identical micro and mainframe (presumably IBM) implementations. The package includes communications software so that programs developed locally can be sent up to a big machine where they will execute without change. This kind of synergism is a quantum jump beyond simple time-sharing as provided by The Source et al.; it makes the responsiveness of a personal computer and the awesome abilities of a big machine equally and simultaneously accessible.

One caution: while Commodore aims for delivery in "late 1981," it seems likely that such an ambitious package could suffer deadline slippage. Still, universities that are planning their computing moves for 1982 and beyond would do well to investigate the "Micro-Mainframe."

PMC Loads Down

Personal Micro Computers makes the PMC-80, which they call "a work-alike to the TRS-80 Model I." They've developed a simple local network system that,



A PMC-80, configured as the master station in a network.

they hope, will make it easy to fill a classroom with PMC-80's while eliminating the hassles involved in loading cassette software. The system is composed of a master PMC-80 with disks and a printer, connected by a simple two-wire link to a number of PMC-80's with more modest options. The master system broadcasts a series of programs; the user of a slave system captures the one wanted as it comes by in its turn. The whole is said to make a useful and economical small network.

CompuServe Reaches Out

CompuServe writes to tell us that their information service now has more than 10,000 registered subscribers, exclusive of demonstration and employee accounts, and serves more than 260 cities.

Atari, Commodore, and Dow Jones

Micro makers have adopted the mainframer's habit of making their big announcements at the time of the National Computer Conference. Commodore timed their "Micro-Mainframe" for the NCC, and Atari came forth with a bundle of announcements as well. Both Commodore and Atari announced access to the Dow Jones Investment Services database. The packages link the personal computer via a telephone to Dow Jones' database of 6,000 stocks and financial news items.

The Atari "Dow Jones Investment Evaluator" program requires an Atari 800 with 48K of storage, a 810 disk, 850 interface, and 830 Modem. The program costs \$100.

The Commodore "Dow Jones Portfolio Management System" runs on a PET 2001 or Commodore 4032 or 8032 with 32K, dual disk drive unit, and modem. Commodore wants \$150 for their software.

Atari and Purser

For some time, Purser's Magazine has performed the valuable service of reviewing and advertising personal software. Robert Purser has now produced a "Special Atari edition" of the magazine and, although the publication has clearly received a lot of patronage from Atari, Purser seems to have maintained his edi-

torial independence. The edition contains two illuminating articles by Chris Crawford, head of the Atari Software Development Group, and reviews of 44 Atari programs.

The "Special Atari Edition" is supposed to be available only through Atari dealers. The Atari dealers in our area carry a remarkably limited stock of software and seem unaware of much software beyond Star Raiders and Super Breakout. If you have trouble finding software, get the "Special Edition." If you can't find that, write to the magazine.

One valuable piece of information we got from Purser's magazine was the fact that Atari have set up the Atari Customer Service center with toll-free numbers. The center is said to be ready to answer "any Atari-related question," and to take orders for the newly-published Atari technical manuals.

More Information Sources

It is ironic that the computer world turns on the printed word. All of us are constantly seeking printed information, while personal computers are used most heavily for word processing. Will computers drag us back to literacy? At any rate, the press release pile gives notice of a few sources of print matter you might not have heard of, including a new journal, another software catalog, and a publisher of useful books.

The new *Journal of Computers in Science Teaching* aims to be "a forum for the exchange of information about teaching science with computers." It is published by the Association for Computers in Science Teaching with backing from the University of Texas. A year's subscription (four issues) costs \$7. The editor is Gary Marks, and he's looking for papers right now.

Creative Discount Software — strange name, that; how creative can a discount be? — is a software retailing firm. Their release affords an interesting glimpse into the way the software market is developing: "the firm . . . has just completed its pre-season major software purchases," they say. We picture the buyers, venturing into the quaint native bazaars, looking

for the unique handcrafted diskettes . . . but since when has software had a season? Anyway, the firm has two catalogs of professional and educational software ready to mail.

Prentice-Hall, under its Spectrum Books logo, has released a list of calculator and computer books. Here are some of the more interesting titles (with paperback prices). *Scientific Analysis for Programmable Calculators* by Harold R. Meck (\$7.95) covers math analysis for the TI 58 and 59 calculators. *Calculator Clout* by Maurice D. Weir (\$8.95) doesn't specify the calculator of interest, but aims at programming methods in business as well as science. Paul Friedman's *Computer Programs in BASIC* (\$10.95) claims to be "a fully indexed guide to over 1,600 BASIC programs recently published in easily accessible personal computer magazines." *TRS-80 Assembly Language* by Herbert Howe, Jr. (\$9.95) aims to teach "all of the information anyone needs to develop machine language programs," a fairly tall claim.

Paddle a Cool Apple

Two Apple add-ons came to our attention recently. R. H. Electronics told us of their "Super Fan," a quiet cooling fan in an Apple-colored box that hitches to the left side of the Apple case. The maker claims that Apples with more than a few circuit cards can suffer from unnoticed overheating, leading to transient problems that are hard to trace. The \$60 "Super Fan" pulls air through the case, lowering the temperature of the circuits.

Peripherals Plus have produced industrial quality joysticks and game paddles for the Apple. Both units use a linear potentiometer that "provides linear control within 1/10 of 1%" and are packaged in metal cases. A set of two "Super Paddles" costs \$40; a "Super Joy Stick" costs \$60.

Software for the Professions

Two items relating the profession of Dentistry to small computers came our way recently. The *Dental Computer Newsletter* is an international publication for "dentists, physicians, and office management people who have interests in office computers." They don't give the publication interval, but a year's subscription is \$12.

Meanwhile, Cybersoft has produced version 5 of their Dental Billing System, claimed to be "the most comprehensive software system available for dental practices." The package consists of 29 programs written in CBASIC for CP/M systems; all functions are accessed through menus. It handles group practices, keeps

detailed patient histories and financial accounts, prints insurance forms, and prepares billings and various reports. We note that the package includes several programs to recover the data base after a system crash or power failure, an essential item that is often overlooked in small computer software. The one-time license fee of \$2000 includes complete source code.

REAL-TABS is an accounting and business information system for the North Star computers. Vincent D. Puzar claims it is good for any type of business but is especially designed for those who work with "the brokerage or leasing of . . . real estate." In particular, the system is said to handle up to 100 escrow accounts. It incorporates a security clearance so that confidential reports remain so, and the author claims "no bookkeeping input, once posted, can ever be tampered with." The system costs \$175.

Finally, High Technology offers a Job Costing System for the Apple II with Pascal. The system can track up to 400 jobs in progress, accepting postings of all sorts of costs and producing reports that let management track progress, fine-tune estimates, and check profit and loss by job.

Software for the Private Party

Those having an interest in genealogy and access to a CP/M or HDOS system might be interested in ROOTS/M and ROOTS89 from COMMSOFT. The programs use professional genealogical conventions to store information on "a virtually unlimited number of relatives and ancestors." Pedigree charts can be displayed for any individual in the file, and the relationship between any two individuals can be calculated.

Hams (radio amateurs, not actors) with TRS-80's Model I or III can now go on the air via the computer with minimum fuss. Macrotronics produces "Terminal," a package of hardware and software that "allows the operator to be on the air receiving and transmitting Morse or RTTY in minutes." The hardware box plugs into the receiver headphone jack and into the transmitter CW key jack or Morse, or into the microphone connector to send FSK tones. The software brings it all to the screen and keyboard. Printer output is supported.

Finally, users of practically any Apple II now have access to what is claimed as "the first true compiler" for that machine. MicroMedia told us about the Applesoft Compiler, written by Jonathan Fiten and "published by the Hayden Book Company," so we're not quite clear on whose product it is. Regardless, the compiler takes a source pro-

gram in Applesoft BASIC and produces 6502 machine code from it. The object program is claimed to run up to ten times faster than the original, interpreted version. The compiler is priced at \$200.

Turn On With TRS-80

Just before deadline, we heard from Radio Shack about their new Plug 'n Power Controller. The controller connects to the cassette jack of a TRS-80 Model I, III, and Color Computer. It operates under the control of the software that accompanies it, and can direct as many as 256 different remote modules to turn on, turn off, or dim. The principle used would be the same as that employed by the BSR Home Controller (which many people have modified for computer control) but there is no indication that Plug 'n Power's signals are compatible with the BSR device. However, the controller is reasonably priced at \$40. The remote receivers won't break you, either, at \$17 or \$18 each depending on load capacity.

Contact Points

Association for Computers in Science Teaching, POB 4825, Austin, TX 78765.

Atari, 1265 Borregas Avenue, Sunnyvale, CA 94086.

Atari Customer Services, 1346 Bordeaux Drive, Sunnyvale, CA 94086; toll-free calls welcomed at 800-538-8543 outside CA, 800-672-1404 inside it.

Commodore Business Machines Computer System Division, 681 Moore Road, King of Prussia, PA 19406; (215) 337-7100.

COMMSOFT, 665 Maybell Avenue, Palo Alto, CA 94306; (415) 493-2184.

Creative Discount Software, 256 South Robertson, Suite 2156, Beverly Hills, CA 90211; toll-free orders through Operator 831: 800-824-7888 outside CA, 800-852-7777 in it.

Cybersoft, 433 Valley Avenue, Birmingham, AL 35209.

Dental Computer Newsletter, E. J. Neiburger Editor, 1000 North Ave., Waukegan, IL 60085.

High Technology Software Products, POB 14665, Oklahoma City, OK 73313; (405) 840-9900.

Macrotronics, 1125 N. Golden State Blvd., Turlock, CA 95380; (209) 667-2888.

Micromedia, 61 South Lake Avenue, Pasadena, CA 91106; (213) 795-9646.

Peripherals Plus, 39 East Hanover Ave., Morris Plains, NJ 07950; (201) 540-0445.

Vincent D. Puzar, 5905 Gulf Blvd., St. Pete Beach, FL 33706; (813) 360-7577.

R. H. Electronics, 3125 19th Street, Suite 173, Bakersfield, CA 93301.

Spectrum Books, Prentice-Hall Inc., Englewood Cliffs, NJ 07632.

NUMBER SYSTEMS

by Alfred J. Bruey



Too many of us have been confused by complicated and drawn-out explanations of number systems. It is a subject most of us prefer to forget as quickly as possible! But in the world of computing, if one wants to get involved with more than a superficial level of programming, he soon runs into binary, octal and hexadecimal notation. Author Bruey presents us with this article which gives a clear explanation of the subject. Keep it on file - even if you don't need it now, there is certain to be a time when you will find it handy.

The number system we use in our day-to-day work is the decimal, or base 10, system. When we use a number, for example 1846.32, a place-value system is implied. That is:

$$1846.32 = 1*10^3 + 8*10^2 + 4*10^1 + 6*10^0 + 3*10^{-1} + 2*10^{-2}$$

The base 10 system is generally accepted by the industrial nations of the world as the common number base.

The early designers of computer systems, those responsible for integrating the hardware and software aspects, found that the most economical approach was through the use of switching circuits. These circuits could ultimately be reduced to a series of "on-off" decisions; that is, a switch was either on or off, there was a hole in a card at a particular place or else there wasn't, or a relay was either open or closed. Because of these hardware considerations, computers were (and still are) designed to work with binary numbers.

While the decimal system uses 10 different digits (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) to represent all numbers, the binary system requires only two. For convenience, these two symbols were chosen to be the digits 0 and 1. The value of a string of these binary digits, or bits as they are called, is determined in the same way as our decimal (base 10) example.

$$1101.11_2 = 1*2^3 + 1*2^2 + 0*2^1 + 1*2^0 + 1*2^{-1} + 1*2^{-2} = 8 + 4 + 0 + 1 + .5 + .25 = 13.75$$

There are three things for you to notice in this example:

1. The use of the subscript to represent the number system base, i.e. 110_2 represents a base 2 (binary) number and 110_{10} represents a base 10 (decimal) number.
2. The point (.) in the bit string example is called a binary point, not a decimal point.
3. The example above shows the method of converting a binary number to a decimal number.

Since we do not want to be required to do all our arith-

metic in binary when we are preparing data for the computer, we must find a way to represent our decimal digits as groups, or strings, of bits. There are several ways this can be done. The most obvious way is to convert each decimal number into its binary equivalent according to the following table:

Base 10	Base 2
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Then we could represent the number 347 as

3	4	7
0011	0100	0111

The bit string that results from representing decimal numbers by converting each digit individually to its binary equivalent is used in some computers, especially for COBOL or assembly language programming. This representation of a decimal number as a bit string that can be used by the computer is called *binary-coded-decimal* (BCD) representation. This coding scheme has one major shortcoming: if you examine the base 10 to base 2 conversion table above, you will notice there are some combinations of bits that never occur. For example, the string 1110 does not represent any digit in the BCD scheme.

One way to eliminate the problem of wasting such bit combinations is to drop the digits 8 and 9 from our table, leaving us with a table of base 8 (octal) conversions:

Base 8	Base 2
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Now all possible combinations of three bits are used to represent a number, but they can only represent base 8 (octal) digits. Therefore, before converting our number, for example 347, to binary, we must first convert it to octal. This is accomplished by repeated divisions by eight, as follows.

0	remainder 5
8	5 remainder 3 = 533 ₈
8	43 remainder 3
8	347

Therefore, $347_{10} = 533_8$, which can be checked by expanding:

$$533_8 = 5*8^2 + 3*8^1 + 3*8^0 = 320 + 24 + 3 = 347_{10}$$

We can now convert 533 to binary the same way that we converted 347 to BCD:

5	3	3
101	011	011

This gives us a true binary representation of the number 347, since we see that

$$101011011_2 = 256 + 64 + 16 + 8 + 2 + 1 = 347_{10}$$

With this representation, we have no unused bit combinations.

A further look into the base 10-base 2 conversion table might lead us to investigate a way to use the 6 unused four-bit combinations. To do this, we could need to define symbols to take us from 10 to 15, since 15 is the highest number that can be represented with four bits, i.e., $15_{10} = 1111_2$.

We will arbitrarily choose the letters A, B, C, D, E, and F to be our additional symbols, so the table will look like:

Base 16	Base 2	Base 16	Base 2
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

This base 16 system, in quite common use today, is known as the *hexadecimal* system. This is another way to represent our decimal numbers with no waste of bit combinations. We use repeated divisions to convert from base 10 to base 16, just as we did to convert from base 10 to base 8:

0	remainder 1
16	1 remainder 5 = 15B ₁₆
16	21 remainder B
16	347

so $347_{10} = 15B_{16}$. Now we can represent our base 16 number digit-by-digit as a binary number just as we did in the BCD and Octal cases:

1	5	B
0001	0101	1011

Note that the binary representation is the same as it was in the octal case, except for the leading zeroes in the base 16 case. The base 16 representation is therefore a true binary representation. Also note that, using powers of 16 as we used powers of 2 and eight in the binary and octal cases, we convert 15B to decimal as

$$15B_{16} = 1*16^2 + 5*16^1 + 11*16^0 = 347_{10}$$

In summary:

1. To convert from base 10 to base x, keep the remainders from repeated divisions by X. These remainders, in the reverse order, will be the required number in base x.
2. To convert from binary to octal or hexadecimal, divide the bits into groups of either three (for octal) or four (for hexadecimal). Then convert the groups to the proper octal or hexadecimal character.
3. To convert from octal or hexadecimal to decimal, use the method of powers of eight or sixteen as shown in the examples. ■

Classifieds

EDUCATIONAL SOFTWARE

PET "Talks" through audio tapes teaching BASIC and simple PET programming for elementary students. Written by experienced classroom teacher. Used two years in classroom with great success. Individualized - minimal supervision by teacher required. Suitable for grades 3 to 9. Activity sheets included. \$57.50 for 5 tapes (10 lessons) and activity sheets. California residents please add 6% sales tax. Frances I. Putnam, P. O. Box 605, Chula Vista, CA 92010.

GAMES

BEST CROSSWORD GAMES ON TAPE!
BEAT PAR-21: For Kids 7-12. Encourages dictionary exploration, builds vocabulary. Two levels of play . . . \$19.95
BEAT PAR-36: For ages 12-adult. Any number can play. Exasperating sometimes, fun at all times! . . . \$24.95
BEAT PAR-72: A challenging version for crossword fans who know you don't visit animals at a XU! . . . \$29.95
 All tapes are TRS-80 Level II, Models I and III.

COD Orders: 800/227-1617, ext 591. In California call: 800/772-3545 ext 591.
 MAIL ORDERS: Rollingwood Publica-

tions, 1603 Solano, POB 7108, Berkeley, CA 94707.

PUBLICATIONS

MICRO MOONLIGHTER NEWSLETTER is the only publication devoted exclusively to aiding YOU in the creation, building and maintenance of your own home-based business using your micro-system. \$25/year U.S., \$29/yr Canada from the publisher: J. Norman Goode, 2115-DD Bernard Ave., Nashville, TN 37212.

SOFTWARE FOR SALE

Professional FORTH development system for the Apple. Includes FORTH Kernel, memory to memory meta compiler, 6502 assembler, editor, file system (supports ISAM, Hashed and Random access files) Name and link stripping vocabulary. Corvus drive I/O drivers. Note: All source code included. No royalties. International Computers; P.O. Box 397, Mt. Arlington, NJ 07856; 201/663-1580 after 6:00 P.M.

POSITIONS AVAILABLE

Microprocessor Programmers and Software Engineers for Santa Clara Valley

Electronic Firms. Please contact Merrie Vander Mel & Assoc., Inc. (a specialized recruiting firm for technical professionals). 3333 Bowers Avenue #130, Santa Clara, CA 95051 (408) 727-1855.

!NEW!

Classified Ad Space Available. Write to: Recreational Computing, Box E, Menlo Park, CA 94025.

Advertisers Index

ASAP.	4
ASCII.	Inside Front Cover
Avant Garde Creations.	46
Harry Briley.	41
Classifieds.	46
Computing Teacher.	39
CRM/McGraw-Hill.	23
Dilithium Press.	47
Electronics Specialists.	22
Huntington Computing.	29
Imprint Software.	23
Information Unlimited Software.	Outside Back Cover
J & S Software.	39
Krell Software.	15
Micro Learningware.	31
Monument Computer.	41
Pacific Exchanges.	7
Dan Paymer.	22
Theisis.	37



A WIDE SPECTRUM OF TITLES

Computers for Everybody

Jerry Willis and Merl Miller
 This fun-to-read book covers all the things you should know about computers. If you're anxious to buy one, use one or just want to find out about them, read this book first.
 ISBN 0-918398-49-5 \$4.95

Microsoft FORTRAN

Paul M. Chirlian
 Here is the the book for you microcomputer users who want to implement FORTRAN on your machines. Even if you've never used FORTRAN before, you will be writing and running FORTRAN programs almost at once.
 ISBN 0-918398-46-0 \$14.95

How to Get Started with CP/M

Carl Townsend
 One of the world's most popular operating systems is explained in simple terms. Includes a handy guide on shopping for an operating system, a glossary, a list of hardware manufacturers supporting CP/M and a list of major CP/M software.
 ISBN 0-918398-32-0 \$9.95

Take AIM: Volume One

James H. Clark
 This lab and learning manual for the AIM-65 and other 6502 microcomputers includes computer precautions, programming basics, and 30 fully documented utility and game programs which teach math, data handling, simulation and more.
 ISBN 0-916460-29-0 \$16.95

Year of the Robot

Wayne Chen
 This thought-provoking book illustrates how a robot encroaches upon the turfs of religion, morality and philosophy, teaching us how to behave.
 ISBN 0-918398-50-9 \$7.95

Small Computers for the Small Businessman

Nicholas Rosa and Sharon Rosa
 If you're ever considered a computer for your business but didn't know where to turn, this is the book that will arm you with all the information you'll need to make an intelligent, cost-effective decision.
 ISBN 0-918398-31-2 \$12.95

From dilithium Press

Write or call for free catalog! 800-547-1842

Most bookstores and computer stores carry our books. Call us on our toll free number and we'll tell you the one nearest you.
 dilithium Press, P.O. Box 606, Beaverton, OR 97075

AVANT-GARDE CREATIONS

P.O. Box 30160-N Eugene, OR 97403

ZES

THE ULTIMATE IN COMPUTER AIDED INSTRUCTION

CREATE LESSONS IN ANY SUBJECTS. NO PROGRAMMING KNOWLEDGE NECESSARY. ZES IS NOT A LANGUAGE BUT A COMPLETELY MENU-DRIVEN AUTHORING SYSTEM WITH:

- HI-RES GRAPHICS
- COLOR AND ANIMATION
- CARTESIAN GRAPHS
- HINTS
- COMMENTS
- BRANCHING CAPABILITIES WITH ELABORATE STUDENT RECORD KEEPING

Demo Pkg. \$10.00
 Complete System \$250.00

ZES FOR BUSINESS

- Lets you quickly and easily create computer aided training/instruction modules.
- Great for on-the-job training in all situations where personnel can't be spared for instructional purposes or to supplement the efforts of vocational instructors.
- A picture is worth a thousand words. With ZES you can create diagrams that can be studied at any time as part of instruction.
- May be used to create material for general informational terminals, such as can be used in vocational counseling, etc.

THE COMPLETE MAILING LABEL & FILING SYSTEM

- Binary sorting for 2 second access
- Sorts all files alphabetically or by zip-code and writes alphabetical and zip directories
- Prints out regular mailing labels with optional added lines
- Prints out labels in file-sequence, alphabetical or zip-code order
- Range sorts, field sorts, or character-code sorts individually or concurrently - very dynamic
- User formatted fields
- Prints out sorted 80-column (or more) formatted address lists, phone lists, or user formatted lists, in file, zip code or alphabetical order
- Multiple (or 2 level) COUNT/SORTS

\$75.00