people's computers

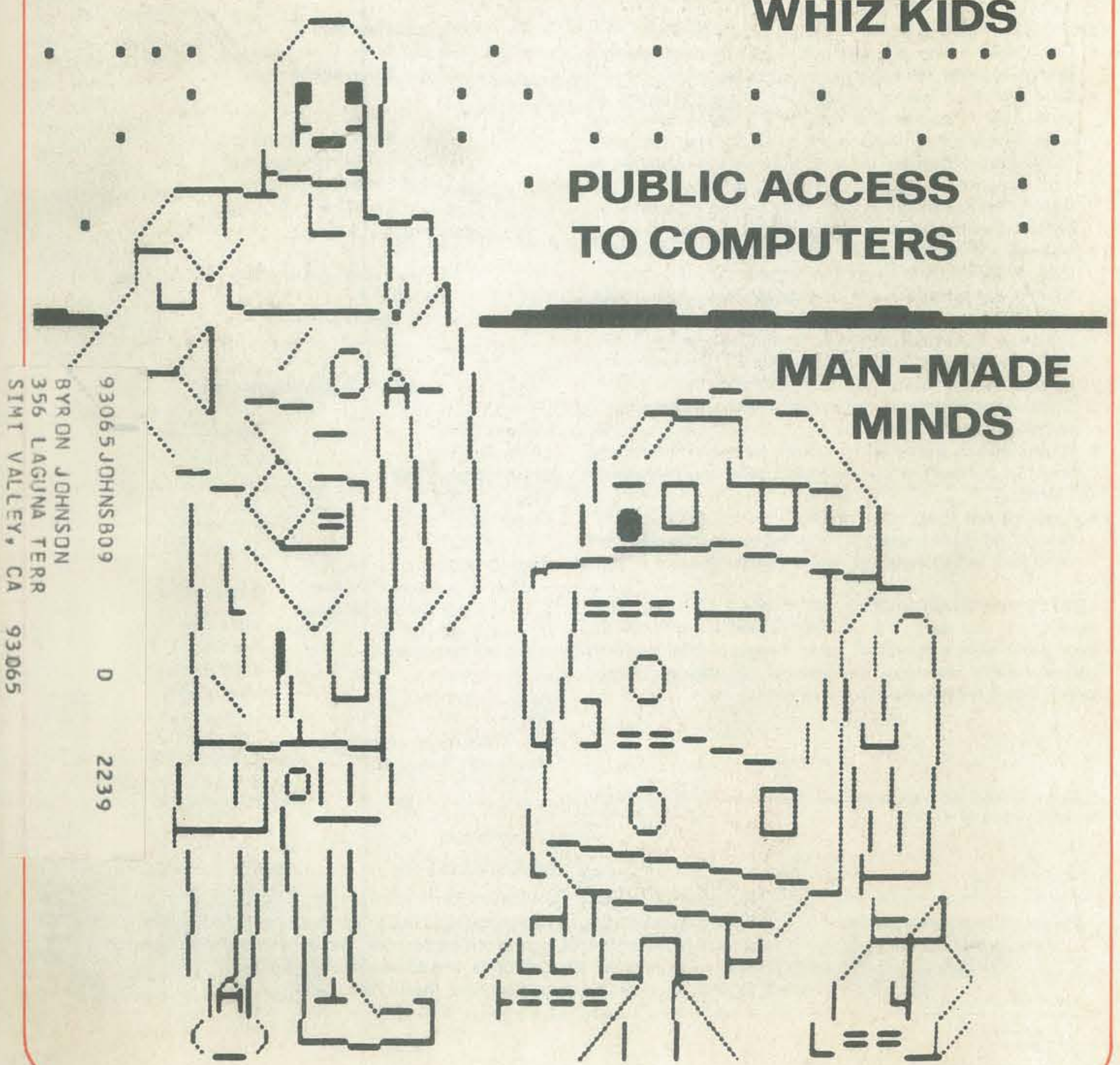
$1.50

VOL 7 NO 1 JULY–AUGUST 1978

COMPUTER WHIZ KIDS

PUBLIC ACCESS TO COMPUTERS

MAN-MADE MINDS



SUBMITTING ITEMS FOR PUBLICATION

LABEL everything please, your name, address and the *date*;

TYPE text if at all possible, double-spaced, on 8½ x 11 inch white paper.

DRAWINGS should be as clear and neat as possible in black ink on white paper.

LISTINGS are hard to reproduce clearly, so please note:

- Use a new ribbon on plain white paper when making a listing; we prefer roll paper or fan-fold paper.
- Send copies of one or more RUNs of your program, to verify that it runs and to provide a sense of how things work – and to motivate more of us to read the code. RUNs should illustrate the main purpose and operation of your program as clearly as possible. Bells, whistles and special features should just be described in the documentation unless they're particularly relevant.
- Make sure your code is well documented – use a separate sheet of paper. Refer to portions of code by line number or label or address please, not by page number. When writing documentation, keep in mind that readers will include beginners and people who may be relatively inexperienced with the language you're using. Helpful documentation/annotation can make your code useful to more people. Documentation should discuss just which cases are covered and which aren't.
- If you send us a program to publish, we reserve the right to annotate it (don't worry, we won't publish it if we don't like it).
- Last but not least, please try to limit the width of your listings: 50-60 characters is ideal. Narrow widths mean less reduction, better readability and better use of space.

LETTERS are always welcome; we assume it's OK to publish them unless you ask us not to. Upon request we will withhold your name from a published letter, but we will not publish correspondence sent to us anonymously. We reserve the right to edit letters for purposes of clarity and brevity.

Cover Illustration: An example of Daniel Browning's ASCII GRAPHIX. See article on pp 40-41.

SUBSCRIPTIONS

U. S. Subscriptions
- ☐ $8/yr. (6 issues)
- ☐ $15/2 yrs. (12 issues)
- ☐ Retaining subscription @ $25 ($17 tax deductible)
- ☐ Sustaining subscription @ $100+ ($92+ tax deductible)

Foreign Surface Mail
- ☐ add $4/yr. for Canada
- ☐ add $5/yr. elsewhere

Foreign Airmail
- ☐ add $8/yr. for Canada
- ☐ add $11/yr. for Europe
- ☐ add $14/yr. elsewhere

Payment must be in U.S. dollars drawn on a U.S. bank.

These back issues are available at $1.50 each:
Vol 5, No 6
Vol 6, Nos 1, 2, 3, 4

Foreign Distributors of *People's Computers*

Vincent Coen
LP Enterprises
313 Kingston Road
Ilford IG1 1PJ
Essex, UK

Rudi Hoess
Electronic Concepts PTY Ltd
Ground Floor Cambridge House
52-58 Clarence St
Sydney NSW 2000

ASCII Publishing
305 HI TORIO
5-6-7 Minami Aoyama
Minato-Ku, Tokyo 107
JAPAN

Eastern Canada

Liz Janik
RS-232
186 Queen St W
Suite 232
Toronto ON M5V 1Z1

Western Canada

Bill Blake
PACOM
4509 Rupert St
Vancouver BC V5R 2J4

Integrated Computer
Greenhills PO Box 483
San Juan, Metro Manila
PHILIPPINES 3113

People's Computers is published bimonthly by People's Computer Company, 1263 El Camino Real, Box E, Menlo Park, CA 94025.
People's Computer Company is a tax-exempt, independent, non-profit corporation, and donations are tax-deductible.
Second class postage paid at Menlo Park, California, and additional entry points.

people's computers

VOL 7 NO 1
JULY-AUG 1978

STAFF

EDITOR
Bob Kahn
ART DIRECTOR
Meredith Ittner
PRODUCTION
Sara Werry
ARTISTS
Matthew Heiler
Maria Kent
Ann Miya
Judith Wasserman
TYPISTS
Maria Kent
Renny Wiggins
BOOKSTORE
Dan Rosset
CIRCULATION
Bill Bruneau
BULK SALES
Christine Botelho
DRAGON EMERITUS
Bob Albrecht
SPOT EDITOR
Phyllis Cole

RETAINING SUBSCRIBERS

David R. Dick
Mark Elgin
John B. Fried
W. A. Kelley
Frank Otsuka
Shelter Institute

SUSTAINING SUBSCRIBERS

BYTE Publications
Paul, Lori and Tom Calhoun
Dick Heiser, The Computer Store

And a special thanks to all the folks at People's Computer Co.: Curtis Abbott, Chuck Bradley, Claire Connor, Delia Daniels, Eric Fulda, Cary Helvey, Bob Jacobsen, Cynthia Kosina, JoAnn Loeffler, Michael Madaj, Drew McCalley, Kathleen Shaw, John Snell, John Strawn, Eryk Vershen, Kim Vose, Tom Williams, Denise Winn.

SPECIAL FEATURES

ARTICLES

MICRO SYSTEMS & APPLICATIONS

DEPARTMENTS

Get TOPO. Map of Lake Alpine
6/22 Water the Plants!!
Check the oil + tires.

EDITOR'S NOTES

Tomorrow we go to press... my first issue! There certainly is a lot of "me" in this issue... perhaps it will spark some reader comment — we've been short on letters lately.

Details!!

- ☐ by lines on whiz kids probs.
- ☐ continuation of letters
- ☐ call N.Y.—where's that perm. letter
- ☐ line nos. on concentration
- ☑ Editor's notes (heh, heh)

Themes this issue:

- Computers + humanity; future intelligence, man-made man
- Computer whiz kids
- Computers + Museums — Public Access
- Educational applications
- Micro systems—esp PET + TRS-80
- Games, fantasy, trivia

THINGS TO COME:

- Artificial Intelligence - what's new? Educational Applications
- Lot's more on languages BASIC Standards Committee
- Designing games + simulations (SLEUTH, KINGDOM, etc.)
- Computers + The arts — musical composition, poetry, graphics (LHS SHAZAM class)
- Color micros — Apple II, Exidy, compucolor, Cromemco, etc.

Note — Check out the word
That Texas Inst. is staffing a
personal computer division.
Also — the $50 T.I. talking
calculators. What's this Fall
TV program in Boston all about?
Need 1/2 doz. eggs, 1 qt. milk, some sourdough bread + some white cheddar cheese.

Bob Kahn

LETTERS

As the owner of a Radio Shack TRS-80, I am entering my subscription with some reservation. I received a copy of your March-April, 1978, issue at the San Jose Computer Faire, and I got the impression that your magazine was promoting the Commodore PET and being less than objective about the TRS-80.

I hope that future issues of your magazine will carry programs and articles that will be of interest to owners of the TRS-80 and other systems in addition to the PET.

William S. Pitt
Pacific Grove, CA

I trust that you received our May-June issue which featured articles on the Heath H-8, Apple II, Video Brain, and Radio Shack's TRS-80 as well as the Commodore PET. It's neither our intention to promote any one particular product nor do we have any particular axes to grind. However, at least right now, we are receiving about three times as many letters, articles and programs concerning the PET as we are for all other self-contained or packaged micro systems. So send us some nifty TRS-80 programs or some useful tidbits for TRS-80 owners—we are delighted to hear from you. —BK

PC's coverage of the TRS-80 seems less than objective. I hope you will do better in the future, e.g., with Level II BASIC, additions, and peripherals. Dan Fylstra's article in *Byte* (April 1978) warrants your attention.

Tom Williams' analysis of personal computing users (in his H-8 article in May-June 1978 PC) seems fallacious in implying that only the hardware hobbyist is interested in programming. It is pretty obvious, even from reading PC, that there is a lot of interest in programming on a packaged microcomputer: witness the enthusiasm for PET, apparently not dependent on canned applications software.

I applaud your efforts to identify the characteristics of an optimum general-purpose programming language. I agree with David Beard: 'it's time to forget about langauges and talk about features.'

Joseph H. Gilbreth
Birmingham, AL

You will indeed read more positive things about the TRS-80 in PC—witness this issue. Also, I don't think Tom Williams ever meant to imply that 'only hardware hobbyists are interested in programming.' Perhaps his distinction between hobbyist and consumer was a little tight, but there is certainly some truth in it. Finally, we'll be publishing lots more regarding desirable features of languages as well as exploring other interesting languages in future issues. —BK

I'm sure this won't be the only letter sent to you in response to Tom Williams' attack on the Radio Shack TRS-80 (PC Mar-Apr 78); It certainly won't be the best or most authoritative. Still, I'd like to have the chance for rebuttal.

Tom really blasted the TRS-80, but when you really begin to analyze what he had to say, you realize that most of his complaints are trivial. So it has a plastic case. So what? What did he expect, cast iron? This case is adequate for its job, which is to keep prying fingers out of the works and to protect them. Tom, did the case on your unit break, crack or otherwise give trouble?

A similar complaint relates to the lack of keyboard rollover. Now, I know I'm not much of a typist, but I didn't even know the keyboard didn't have rollover till I saw Radio Shack's ad for Level II BASIC, which does. Even if you're an experienced touch typist, I doubt you'll be able to type BASIC instructions fast enough to ever miss a character.

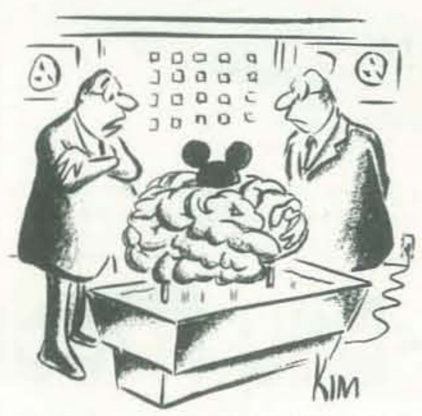
Another 'major' complaint was related to the separate components used in the TRS-80, as opposed to the unit construction of the PET. Really, isn't this a matter of personal preference? There are pros and cons for both approaches. I'll admit that three power cords is an awkward number, but extension cords are cheap. The separate keyboard does permit a more comfortable typing position, just as R-S claims. It may be a drag to disassemble the system when you want to move it; but I'd hate to try to carry it in once piece!

As for the Level I BASIC . . . O.K., everyone concedes that it's crude. On the other hand, it's a super BASIC for 4K. A little while back, PC was touting Palo Alto Tiny BASIC as the greatest thing since Wheaties. What changed your mind?

Finally, Tom refers to the limited graphics capability of the TRS-80. I don't know how to tell you this, Tom, but it has the same resolution as the PET. The difference is that Radio Shack elected to split up the screen into separately controllable pixels, whereas Commodore chose to use special graphics characters. The trade is better looking graphs against real resolution. O.K. if you want to play Blackjack, but rotten for bar-graphs.

So much for the judgement calls. Now for the real inaccuracies. I don't know how Tom managed to get the cassette input to screw up. When I got mine, it took a couple of tries to convince me that you really have to set the volume control between 7 and 8, just like R-S said. After that, the tape I/O has been dead reliable. We have just finished a three-month course in which my computer and one other were used by students, four hours per week. To my knowledge, no tape problems occurred, and that includes transferring tapes between two machines. Try that on two PETs, Tom.

I also can't understand Tom's complaint about the CRT resolution. The 12" monitor is based upon a commercial unit (did you expect a custom one?) However, the video circuits clearly have been modified. The resolution, when the controls are properly set, is quite adequate for 64 characters. When comparing it with a Sanyo or PET, you should bear in mind the 50% larger screen. The apparent resolution of small screens is well known, but an illusion.



"I think we should have concentrated a bit more on its emotional development."

Human nature is a funny thing . . . when one buys a certain brand, he always seems to dislike the others. Thus the Ford/Chevy arguments. Tell the truth now . . . does Tom Williams own a PET? Do I like a PET? No. Why not? 'Cause I'm human, too.

Dr. Jack W. Crenshaw
Huntsville, AL

Tom Williams responds: *First, the system under discussion was the 4K Level I system for $599. It does seem that the 16K Level II system is a better value, but we have not had a chance to check it out. As to the points you raise, the little plastic access door in the rear of the unit has fallen off and is long since lost. I am not an experienced touch typist, but I found the lack of rollover extremely annoying. It was not the 'limited graphics' I was attacking so much as the manner in which they are advertized. Five people (experienced users, all) had trouble with the tape system, and for the $199 stand-alone price I certainly* do *expect more from the monitor than a modified TV set. —TW*

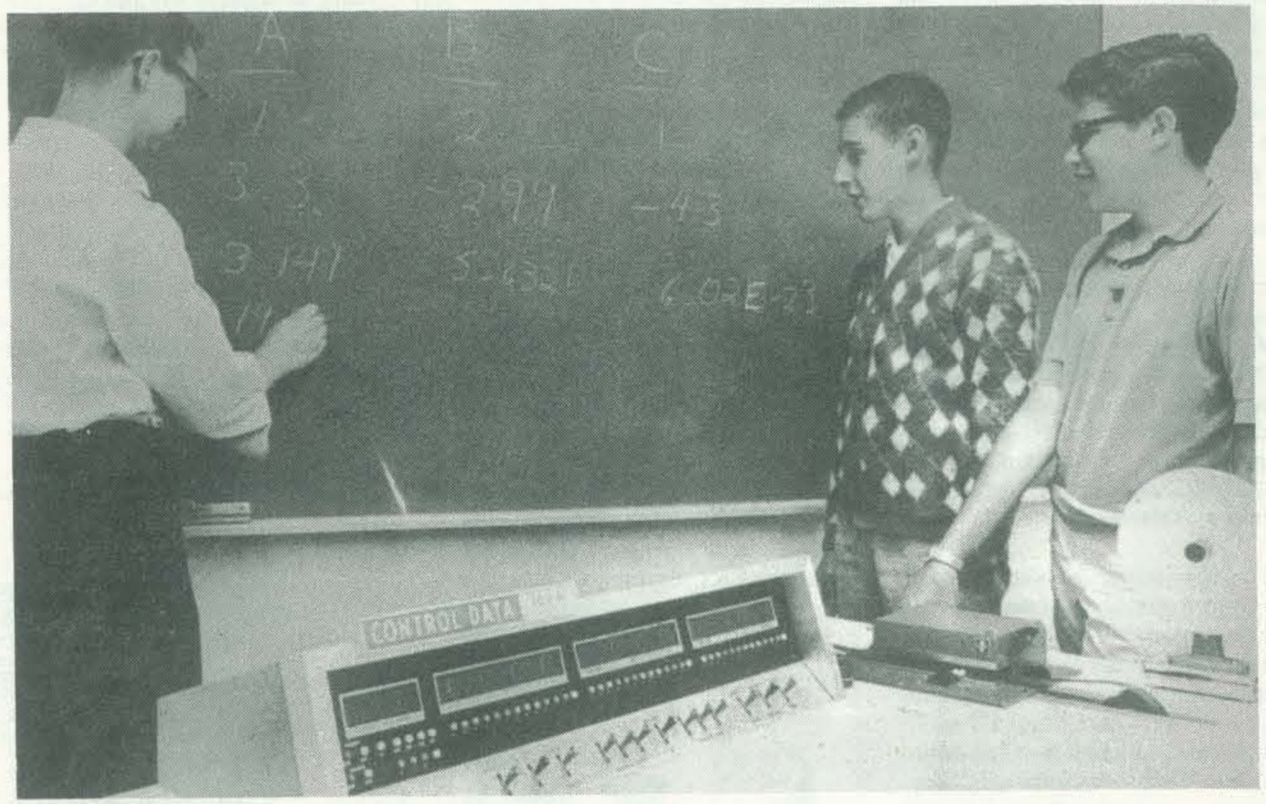
In the June '78 issue of *Kilobaud* Wayne Green is puzzled by the *People's Computers* review of the TRS-80: '. . . the recent blast in PCC at the (TRS-)80 is a mystery . . .' For me the mystery is the way Wayne and Radio Shack have ignored the substantive issues raised in the PC review, such as:

- The cassette recorder operates at a slower speed than most similar equipment. It (to put it charitably) is awkward to use.
- The Level I BASIC is slow and primitive.
- The keyboard has no rollover.

I'm glad to see that PC is meeting its obligation to readers by printing the straight story about equipment, warts and all. Reviews pointing out equipment deficiencies are much more valuable than 'gee whiz' puffery taken from manufacturer's advertising brochures. Perhaps the difference between *Kilobaud* and PC treatment stems in part from the fact that *Kilobaud* runs full page ads for the TRS-80 and PC has no ads at all. Keep up the good work.

Dave Caulkins
Los Altos, CA

Continued on p.28.



Grinding up roots for a medicine show: 'Barker' (and future PEOPLE'S COMPUTERS Editor) Bob Kahn (left) instructs two eager novitiates in the proper feeding of the machine.

A Modern-Day Medicine Show

The occasion of my first issue as editor of PEOPLE'S COMPUTERS calls for a bit of nostalgia, since it was Dragon Emeritus Bob Albrecht, founder of this magazine, who first sold me on computers. This story goes back some 16 years, so climb into our time machine, sit back, and I'll set the chronometer to return us to those 'thrilling days of yesteryear. . .'

The scene is Denver, Colorado (a big, small-town located way up at the base of the Rocky Mountains); the year is 1962. Large scale, million-dollar, number-crunching computers that live in windowless basements of fortress-like computer centers are just reaching their heyday, while $90,000, 12-bit 'mini-computers' that fill an office-sized desk and communicate with us humans via 6-channel, punched paper tape are just coming on the scene. The Dragon (known then as Bob Albrecht) is working as an 'applications analyst' for Control Data Corporation and moonlighting as a medicine-man, traveling around to local high schools giving out free introductory lessons in FORTRAN programming. And I (your fearless editor) am an unsuspecting junior at George Washington High School . . . but I'll let the medicine-man tell the story. . .

BY ROBERT L. ALBRECHT

Note: this article first appeared in DATAMATION® Magazine in July, 1963. It is reprinted here with permission—and with the blessing of Ed Yasaki, DATAMATION's long-time San Francisco Bureau Manager who edited this piece for Albrecht 15 years ago. –BK

During the summer of 1962, the Denver, Colorado, office of Control Data Corporation gave a course in the use of computers to a group of gifted mathematics students from Denver's George Washington High School (GWHS). Emphasis was placed on the use of a computer as a computational tool to assist students in solving mathematical problems. We had so much fun that we continued the course at GWHS throughout the 1962-1963 school year, and extended the program to eight other schools in Denver and Jefferson counties. So far, more than 200 students have been introduced to the hardware as a handy device for getting answers to mathematical problems.

On March 24, a CONTROL DATA 160A and a Flexowriter were moved to GWHS and placed in charge of four students. For the next four days, these students put on a computer demonstration which had many of the characteristics of a 'Medicine Show'. The show was promoted, produced, and directed by the four students, who were also the stars.

The medicine show at GWHS was the high point of a computer educational program which began quite by accident. In the spring of 1962, I acquired a new neighbor, Irwin Hoffman, a mathematics teacher at GWHS. Irwin invited me to speak to the Math Club at his school on the use of computers for the solution of mathematical problems. Following this talk, several students inquired about the possibility of receiving computer training. I conferred with Irwin and we decided to organize and teach a course during the summer.

In this initial course, 35 students from GWHS met every Wednesday evening from 7 to 10 P.M. at the Control Data Denver office. We spent the first few meetings discussing computer fundamentals and describing the FORTRAN programming language for the 160A. By midsummer, the students had become reasonably proficient at writing programs and operating the computer. For the rest of the summer, the Wednesday evening meetings were used primarily for the analysis, programming, and computer solution of mathematical problems. Of the 35 students who began the course, 25 were still going strong at the end of the summer.

The summer course was merely an appetizer for many of the students. They clamored for more training. During the fall semester we set up an advanced course for 12 students. (The rest had graduated.) A bleary-eyed mathematics teacher (Hoffman), a bleary-eyed Control Data instructor (me), and 12 bright-eyed young scholars met every Thursday morning from 7:30-8:30 A.M. before regular school hours. This course was conducted on a seminar basis, using the algebra of polynomials as the mathematical subject. The instructors posed problems. The students did the required mathematical analysis, developed problem solving procedures, programmed their solutions in FORTRAN and ran their problems on the 160A. Introductory courses were set up for beginners. The courses were conducted on an auto-instructional basis with seminar students serving as instructors.

In March, we decided to put on a medicine show at GWHS. We decided that it would be done entirely by students without any adult supervision. By this time we had about 40 students at various levels of proficiency. We picked Bob Kahn, Al Nelson, Randy Levine, and Fred Ris to run the show. (Bob, Randy, and Fred were juniors, Al was a senior.) We outlined our project to the students and sat back to watch the fireworks.

During the week before the show, the four students arranged for the use of a room, obtained the equipment and supplies needed, and began an intensive promotional campaign. They drew up a master schedule for the 20 math classes which were to attend, and distributed the schedules to the teachers who were responsible for the classes. After some haggling, they agreed upon a demonstration program, wrote the source program, and checked it out after the 160A had arrived at the school. Articles plugging the show appeared in the school paper, and daily spot announcements over the P.A. system reminded the student body of the forthcoming event. By opening day, our students were indeed prepared.

THE MEDICINE SHOW

The 160A was moved in on the afternoon of Monday, the 24th, and turned over to the students the next morning. The demonstration program, chosen in the hope that it would be meaningful and entertaining to the attendees, was checked out. Their source program, shown in Figure 1, was a procedure to compute the real or complex roots of the quadratic equation, $AX^2 + BX + C = 0$. However, it had a gimmick.

Figure 1

```
0
0
C       GWHS TEACHER DEMO PROGRAM
C       PROGRAMMED BY RANDY LEVINE
1       FORMAT (45Hi ;have ;heard ;you ;do ;not ;like ;assistance ;. ;. ;.)
2       FORMAT  (36Hhere ;is ;your ;iddy ;biddy ;answer. ;. ;. ;)
3       FORMAT  (50Hhere ;is ;ye ;olde ;easy ;answer ;. ;. ;. ;lazy ;tiger ;. ;. ;.)
4       FORMAT  (38Hi ;will ;be ;a ;more ;courteous ;gw ;patriot.)
5       FORMAT  (27Hname ;the ;new ;baby ;after ;me.)
6       FORMAT  (39Hyou ;coach ;. ;. ;. ;let ;me ;teach;the ;class.)
7       FORMAT  (45Hwhan ;that ;aprille ;with ;his ;shoures ;shoote. ;. ;.)
8       FORMAT  (43Hwhen ;i ;consider ;how ;my ;light ;is ;spent ;. ;. ;.)
9       FORMAT  (35Hi ;have ;to ;be ;excused ;from ;the ;room.)
10      FORMAT   (I5, 4E16.8)
11      FORMAT  (4E16.8)
500     READ 10, JWHICH, A, B, C
        GO TO (15, 25, 35, 45, 55, 65, 75, 85, 95, 105), JWHICH
15      PUNCH 1
        GO TO 105
25      PUNCH 2
        GO TO 105
35      PUNCH 3
        GO TO 105
45      PUNCH 4
        GO TO 105
55      PUNCH 5
        GO TO 105
65      PUNCH 6
        GO TO 105
75      PUNCH 7
        PUNCH 8
        GO TO 105
85      PUNCH 8
        GO TO 105
95      PUNCH 9
105     DSCRIM = (B'B) - (4'A'C)
        IF (DSCRIM) 100, 200, 200
200     X1  = (-B + SQRTF(DSCRIM))/(2'A)
        X2  = (-B - SQRTF(DSCRIM))/(2'A)
        PUNCH 11, A, B, C
        PUNCH 11, X1, X2
        PAUSE 500
        GO TO 500
100     X1REAL = (-B)/(2'A)
        X2REAL = X1REAL
        X1IMAG = (SQRTF(DSCRIM))/(2'A)
        X2IMAG = -X1IMAG
        PUNCH 11, A, B, C
        PUNCH 11, X1REAL, X1IMAG, X2REAL, X2IMAG
        PAUSE 500
        GO TO 500
        END
        END
```

By means of a code number entered with the data, the computer could be directed to precede the answers with a Hollerith message. These messages were terms favored by the instructors in their classroom presentations, and which had become cliches familiar to students. During demonstrations, students were asked to provide data for processing; for these results, no Hollerith message was punched. Then the teacher was asked for a set of data; when his results were listed, he was confronted by a familiar phrase preceding the answers. Some samples are shown in Figure 2.

```
4/9./0./-16./ ·
6/5./4./3./
7/1./10./25./
i will be a more courteous gw patriot.
   .90000000e 01   .00000000e-32  -.16000000e 02
   .13333333e 01  -.13333333e 01
you coach . . . let me teach the class.
   .50000000e 01   .40000000e 01   .30000000e 01
  -.40000000e 00   .66332496e 00  -.40000000e 00  -.66332496e 00
whan that aprille with his shoures soote. . .
when i consider how my light is spent . . .
   .10000000e 01   .10000000e 02   .25000000e 02
  -.50000000e 01  -.50000000e 01
```

Figure 2

The formal demonstrations were held Wednesday and Thursday, about every half hour, from 8:45 to 3:30 P.M., with a few left over for Friday morning. Because many of the classes were discussing the quadratic formula, a frequently heard request was, 'Would you please ask the computer to do the odd exercises on page 263 of our text?' Bob Kahn accommodated one class by running one night's homework on the 160A, and using the Flexowriter to cut a Ditto master of the output tape. He presented a copy to each member of the class.

When the hardware was not being used for demonstrations, it was in constant use by students in the computer educational program. They practically fought over it. In fact, they coerced Irwin Hoffman into arriving every morning at 6:30 to let them into the school, and the custodian had to practically throw them out every evening at 6:00 in order to lock up.

The four 'barkers' were excused from classes during most of the show. On Friday morning, Fred Ris rejoined his algebra class just as it began a discussion of the problem, 'Given a set of N people in the room, what is the probability that two of them have their birthdays on the same day of the year?' The class set up some specific cases, and began hand computing the results. Fred jumped up, mumbled something about a computer, and dashed out the room. He ran downstairs, wrote a FORTRAN source program to tabulate the desired probabilities for N = 2 to N = 150, then compiled and executed the program. He managed to get back to his algebra class before the end of the hour to display his results.

Friday noon, the students were informed that they would have to move the computer to make room for a meeting. They rolled the 160A out of the room, down the hall, and set up shop in the main lobby. There, they were immediately surrounded by curious students, and ran informal demonstrations until the middle of the afternoon. At that time, a moving van rolled up to the front door, the gear was packed up, and the medicine show moved on to the next stop.

THE SHOW GETS RESULTS

Well organized and conducted with imagination and showmanship, the performance was a tremendous success. It generated a great deal of interest in additional introductory computer courses. Anticipating this, the boys had already arranged for the use of our classroom facilities and the 160A during spring vacation. Sixty students were signed up during the show, the results of a flyer which they printed and distributed, inviting spectators to enroll in introductory FORTRAN classes.

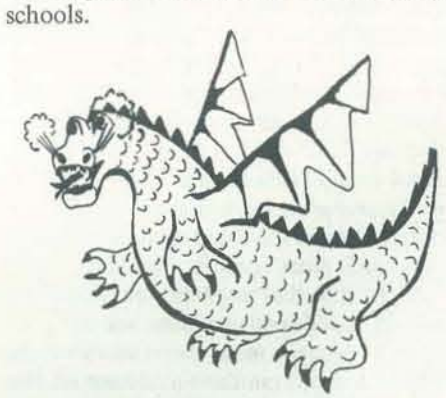
Sessions began a week later, organized into four sections of 15 students each. The courses were run on an auto-instructional basis. We provided 60 copies of a workbook *Introduction to FORTRAN Programming*, which we are writing and reproducing in our office to teach the use of computers for mathematical problem-solving in the secondary schools. For a week, the student instructors answered questions and tutored individuals. During the second week, spring vacation, the instructors brought each section to our office for a day of problem-solving on the 160A. Every student had executed at least one FORTRAN program by the end of the week, and several had run more. They chose their own problems, most of them coming from their mathematics text books.

At Washington High, as a result of this activity, we now have nearly 100 students who will be clamoring for additional training during the next school year.

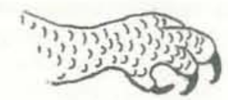
PLANS FOR THE NEXT YEAR

We are sold on medicine shows. In fact, the second show was put on by five students at Denver's Abraham Lincoln High School, and more are anticipated during the next school year (1964).

It will be a year of growth. In adjacent Jefferson County, public school officials are making arrangements to train high school teachers in computer methods, and have invited several universities in the area to assist in implementing an extensive program in computer education. During the next school year, we expect to have courses there in computers and computing, as well as the present courses in problem-solving and answer-getting with FORTRAN. Recognized as a leader in evaluating and adopting new ideas, the Jefferson County public school system's program should provide us with some data for evaluation of several approaches to computer education in the secondary schools.



Editor's Note: During the years 1964-1974, Medicine-man Albrecht underwent a metamorphosis from which he emerged . . . a dragon! So, if the epilog to this story, which was written quite recently, seems to have changed stylistically, it is probably not *your imagination . . .*



EPILOG: OR THE DRAGON'S TOUCH

And so it came to pass that computers found a prominent place in the educational system—and thus in the minds and hearts of children—throughout the mountainous land of Colorado. Last we heard, some 16 years after the original medicine show, GWHS math teacher Irwin Hoffman was still directing one of the most prominent and long-standing high school computer and math education programs in the nation. Indeed, GWHS was included in a 1977 nationwide *Academic Computing Directory*—compiled and published by the Human Resources Research Organization (HumRRO) under the sponsorship of NSF—as one of 106 institutions having exemplary programs in computer education.

Meanwhile, in 1964, the medicine show moved on (as all medicine shows must); Albrecht felt the pull of other worlds and new adventures (a characteristic of medicine-men), and the 'barkers' of the GWHS Medicine Show scattered in the directions of the four winds to various prestigious institutions of higher education.

Albrecht wandered to the north and east, soon arriving in Minnesota, a low, flat land renowned for its great lakes, green fields and mosquitoes. There he discovered a new, interactive programming language, BASIC. For a time, he dallied in the Northstar State, touting the virtues of this new language as a boon companion for children of all ages. Then one cold day, he heard about a warm, mystical land (that is, a land that had more than its share of mystics) far to the West—a land called California.

A dragon at heart, Albrecht was irresistibly drawn to this strange and glorious land where unusual ideas grew like flowers—everywhere available to creatures with questing, imaginative minds. And, little by little, the dragon in his spirit emerged. He soon organized S.H.A.F.T. (The Society to Help Abolish FORTRAN Teaching), and he began to write books that would encourage children of all ages to learn to read and understand BASIC.

And what became of the four GWHS 'barkers'? Alas, little is known about the wanderings and whereabouts of Ris and Nelson. Levine graduated from Berkeley in 1968 and was last reported doing post-doctoral research and teaching astronomy and cosmology at Harvard. Thus, of the four, only Kahn's story is well-chronicled and able to be told here.

Kahn left GWHS and migrated to a small college called Occidental in the south part of the mystical land of California. However, being a spirit of free expression in the mid 1960's, he was naturally drawn north to Berkeley. There, he met up with his old friend Levine . . . and with Albrecht who had just arrived from Colorado via Minnesota. (The medicine-man was almost unrecognizable as he had grown scales and claws and was beginning very much to resemble a dragon.)

Dragon Albrecht started exorcising the FORTRAN spirit from Kahn's mystified memory by paying him to check out all the BASIC programs in the Dragon's newly authored books. Since Kahn was usually short of cash, but desirous of the finer things in life (such as toasted almond ice cream), he was always delighted to find a way to earn a few extra bucks.

And so, Kahn worked his way through college writing programs in BASIC (and occasionally in FORTRAN, just to keep in practice).

In 1972, he became the Director of the Computer Education Project at U C Berkeley's Lawrence Hall of Science, keeping alive the spirit of the medicine show. Indeed, Lawrence Hall also became an exemplary institution in the HumRRO *Academic Computing Directory*. And, to this day, that spirit continues to spiral outward and beyond.

Meanwhile the Dragon started to publish a Newspaper called PCC, and he made Kahn a member of PCC's Board of Trustees; but that is another story. . .

Thus ends the tale of the Modern-day Medicine Show. However, for dragons, every ending is a beginning. The beginning of PCC, for which this story is the end, was chronicled in PCC Newspaper, Volume 3, Number 5, May 1975. There will be more endings, more beginnings. —The Dragon. □

▷▶▷▶▷▶▷▶ NEWS

COMPUTERIZED TAX PROGRAM

Eighth grader wins science fair honors

Fourteen-year-old Ralph Lipe and parents. Ralph took top honors at the recent Fort Worth Regional Science Fair by programming his TRS-80 to go step-by-step through the 1040-A tax form.

FORT WORTH, TX—Eighth grader Ralph Lipe of McLean Middle School in Fort Worth, Texas received top honors in a recent Science Fair for his computer program designed to guide a person, step-by-step, through the 1040-A tax form on a Radio Shack TRS-80 Microcomputer.

It took Ralph, 14, a total of 60 hours to write and de-bug his BASIC program. The idea for the program came from his mother, Jane Lipe, a certified public accountant, who assisted him in setting up some of the equations for the tax questions.

Ralph's project was chosen 'Best of Fair' out of approximately 200 entries in McLean Middle School's intramural science fair. Then, at the Fort Worth Regional Science Fair, which hosted 460 entries from seventeen counties in North Texas, Ralph won a first place ribbon in the Math & Computers category, a first place ribbon from the IEEE, and placed '3rd Best of Show' for the Junior High School division.

Ralph's interest in computers developed last year when his father, Gary Lipe, brought home a TRS-80. When Ralph first sat down at the microcomputer, he knew nothing about making it operate. Within a few hours, however, he'd taught himself to write a few simple programs. Ralph, who someday hopes to run a computer for NASA, thinks computing is 'almost like a sport that everyone can enjoy.' □

CALIFORNIA'S COMPUTER WHIZ KIDS

BY SUZANNE RODRIGUEZ

This article brings me a pleasant sense of deja vu . . . Ah, to be a young 'computer whiz kid' again . . . (sigh). When Dragon Emeritus Albrecht first came to my high school to talk about FORTRAN (see p.6 of this issue), these whiz kids were busy being born! We've come full circle, and I am delighted to publish this article in my first issue of PC.

Suzanne Rodriguez is a freelance writer who will be studying for an advanced degree in journalism at Stanford University starting next Fall. She is currently working at Zilog, Inc., the people who bring us the Z-80 and other amazing chips off the old silicon block . . . Also working at Zilog is Judy Estrin, who along with Barbara Koalkin of XEROX Palo Alto Research Center, organized and managed this computer competition. –BK

Welcome to the JCIT Bay Area Youth Competition in Computer Science.

This competition has been organized to:

* Encourage creative thinking and stimulate the learning of computer science and applications at the high school level.
* Develop a transfer of technology among student participants.
* Further a cultural exchange and a deeper understanding among contestants from the United States, Great Britain, and Israel.

The winning team, Eric Enderton, Alex Melnick and Chris Pettus of Santa Monica will represent the United States this summer in an international high school computer contest in Jerusalem.

Eric Enderton, Alex Melnick and Chris Pettus know quite a lot about computers. They are largely self-taught: only two of them have ever taken a programming course, but because they've had access to a computer lab, they've tinkered and played around with computers and built up a large body of technical know-how over the last four or five years. There's nothing particularly extraordinary about any of this; what *is* extraordinary, though, is that four or five years ago they were only 11 or 12 years old.

All of this early interest in computers has presented the young Santa Monica (CA) High School students with the opportunity to represent the United States this summer in an international high school computer contest in Israel. The contest is one of the highlights of the Third Jerusalem Conference on Information Technology (JCIT III), and the California students will be competing against students from England and Israel for the first prize of $1000.

All together, twenty-four three-person teams competed to become the US representatives at semifinals held at both Stanford University and the University of Southern California. Two teams were chosen from each group, and at the finals held at USC on May 5, Eric, Alex and Chris became the US finalists.

'US finalists' is more or less a loose term –the selection process for the US took place entirely within California. This was due, in part, to limited funding and time available to the US organizers (Prof. Daniel Berry of UCLA, Judy Estrin of Zilog and Barbara Koalkin of Xerox); but it was equally due to the unique role California plays within the microcomputer industry. The ever-growing 'Silicon Valley' has spawned around itself a state full of computer freaks. Virtually all junior colleges and universities have started or greatly expanded their computer science departments in the past decade. De Anza College, located at the epicenter of the Silicon Valley, offered 18 different computer courses this past spring, and some courses, like beginning BASIC, were offered in 20 different time slots. Many high schools in California now offer programming classes. Some, like Santa Monica High School, even have sophisticated computer labs. Probably no other state approaches California in the myriad of computer facilities, classes, expertise and job opportunities available to its citizens.

The end result of all this–as far as the US/California contest was concerned–was a group of youngsters ranging in age from 15 to 18 that left the judges stunned. The teams had to solve a three-hour programming problem, and most of the students felt the problems were too easy. In a College Bowl-type question and answer session, the teams were pitted against each other, and answers to complex questions involving boolean algebra, algorithms, programming and computer history were often answered before the question itself had been completely posed.

'I think we should hire these kids and get a jump on our competitors,' one of the judges said, shaking his head with amazement. 'They're so quick!'

The judges weren't the only ones who thought highly of the contestants: the contestants themselves were filled with enthusiasm for their own abilities. When asked what they thought of the questions, almost all the students agreed they were too easy. A few written comments were very much to the point:

- '. . . Since you are dealing with the top students in the area, problems must be challenging enough to discriminate among the top teams. . .'
- 'Here's how I would rate the questions they gave us: [4*3-4/8] (in other words, average).'
- 'Questions requiring rote memorization should be left to the computers.'

In early August the contestants will be off to Israel for JCIT III. The theme of this year's conference is the transfer of

SAMPLE PROGRAMMING PROBLEM

The following is a short programming problem worth 50 points. Included in your solution should be a description of your algorithm as well as the actual coded program.

Chocoholics Delight

There are N pieces of chocolate arranged in a circle. Beginning at a particular position, we count around the circle and eat every Mth piece. For example, the order in which the chocolate pieces are eaten when N=8 and M=4 is 4, 8, 5, 2, 1, 3, 7, 6 (i.e. the 4th piece is eaten first, the 8th piece second and the 6th piece last. Try this out and see for yourself). Write a complete program which prints out the order in which the pieces are eaten when N=24 and M=11 starting from 1. State the data structures used, their initialization and the program.

–Adapted from Donald E. Knuth's Josephus Problem

SAMPLE PROGRAMMING PROBLEM

Following is a programming problem worth 100 points. Included in your solution should be a description of your algorithm as well as the actual coded program.

Help end the heartbreak of ringworm!! We'll use a square two dimensional array of INTEGER memory cells to represent a patch of skin. Each cell (element in the array) is either healthy, sick, or immune. In order to operate, the program needs three numbers. The first, an INTEGER which we'll call SICK, tells how many time steps a skin spot remains sick once it has been infected. The second, an INTEGER called IMMUNE, tells how many time steps a skin spot remains immune after it is through being sick. The third is a REAL named RATE which tells the odds (probability) that a sick spot will infect a neighboring healthy spot during the current time step.

To represent a healthy spot of skin, store a 0 in the corresponding array location. To start things off, make the whole patch of skin healthy except for the spot in the very center. Store a 1 in the location representing a skin spot when the spot becomes sick. For each time step, sweep through the entire array. At each position with a non-zero value, add 1 to reflect the passage of one unit of time. Then check to see if the value stored there is greater than SICK+IMMUNE). If it is, restore that spot to the healthy state (0).

Next, still within the same time step, sweep through the array again. This time, stop at each spot that is sick and see if it infects any of its healthy neighbors.

You might use an auxilliary array so that the states of affairs at time step t and t+1 don't get confused. You also can convert INTEGERs into characters to make the output nice. We suggest a blank for healthy spots, a '*' for sick spots and a '.' for spots in the immune state.

This program is an interesting one to actually implement. You might try it when you get back to school. By trying different values of the three numbers SICK, IMMUNE and RATE, you can observe a variety of 'diseases', ringworms, blotches, measles, infestations that die out (cure themselves) and ones that continue re-infecting recovered skin.

–Problem by William M. McKeeman

San Francisco Bay Area Semi-finals. Judges (left) confer on solutions of Ygnacio Valley team (middle) and Saratoga team (right) as contest organizer Judy Estrin looks on.

information technology – a somewhat stately bow to the immense technological transfer from other countries which has facilitated Israel's rapid growth and entry into the modern world. Israel recently celebrated its 30th birthday and, as its second generation comes of age, the country's need for technological transfer is beginning to lessen. It is perhaps symbolic that these young American and British representatives go to Israel this summer not to instruct but to compete, on a level of equality, with their technically-minded counterparts.

Symbolism aside, though, this is a great adventure for three of California's computer whiz kids, probably the first of many opportunities that will present themselves through working with computers. 'The furthest I've been away from home is Encinada,' said Eric. 'I just don't believe this is happening.' □

H-8 REVISITED

BY TOM WILLIAMS

In my article on the Heath H-8 computer system which appeared in the previous issue of *People's Computers*, I was critical of the 'command completion' feature incorporated in Heath software. This is the function which finishes typing a command as soon as the computer recognizes the character string entered as being unique. Thus, typing DU will cause the computer to immediately finish the command, DUMP, with no more entry by the user.

It turns out that after using Heath software for a time, one becomes quite used to command completion and thinks of the commands in terms of their abbreviated forms–the ones that trigger command completion. The main advantage of the feature, however, is that in the command mode, it does not allow you to enter unacceptable commands. Thus, after typing L, the computer will not accept the character, Y, but only O (for LOAD) or E (for LEN). Once you have gotten over the habit of typing the whole commands, it's a lot of fun to see the computer anticipate you.

Another nice touch is that Benton Harbor BASIC gives you a chance to reconsider before you execute a command that would destroy the contents of program memory. LOAD <cr> is followed by the H-8 printing SURE?. This gives you another chance to see if you've pushed the 'play' button on the cassette recorder or if in your heart you really want to erase what is currently in memory to load something else. The same 'last chance' is given for the SCRATCH command. Simply typing Y in response to SURE? will cause the H-8 to execute LOAD or SCRATCH.

The updated version of Heath's Extended BASIC contains three new types of files: one for BASIC programs, one for BASIC data only, and one for programs and data. It is thus possible to store the program variables on tape and thus use different sets of starting variables for the same program. The only problem is you can't chain modules–that is, get to different sets of data from within a program. Given this same limitation, you can also load and dump program text only, or you can save both the program and the symbol table. There is a provision in the BASIC that allows the user to read from the ever-versatile H-8 front panel display which type of file is being read. There seems to be a vast number of monitoring possibilities for the front panel and ambitious programmers should keep that in mind when writing that super program.

One more feature I've noticed is that the updated version of Extended BASIC has a provision allowing you to load programs recorded in the previous version (note: this applies to Extended BASIC only). Simply type OLDLOAD <filename> and the new version will gobble up your old Wumpus! □

ROBOTS in LITERATURE and in LIFE

BY LORETTA GRAZIANO

Loretta Graziano is an economist, holding a master's degree in international economics from The Fletcher School of Law and Diplomacy at Tufts University, and specializing in the field of international trade. Her interest in machine intelligence and man-machine relationships started during her undergraduate years at Cornell University where she used to frequent the computer center, chatting with engineering and computer science students.

During her senior year at Cornell, Loretta decided formally to expand her knowledge about computers; she thus enrolled in some introductory computer science courses, including one dealing with the social aspects and implications of computers. This article grew out of reading and research for that course. –BK

How far do you let your imagination go when you dream about robots? Do you go all the way, envisioning an artificial intelligence Shangri-la? Or do you have foreboding premonitions of a technology-gone-wild robot society? Whichever, you are no doubt giving increasing thought to the impact of artificial intelligence, or creative thinking by machines, now that such prospect is more enticing, if not more imminent.

What would be the role of artificial intelligence in the world of humans? Or more incisively, what would be the role of humans in a world of robots? Many literary works deal with this realm of thought, and in their musings raise the question of where the dividing line between human intelligence and machine intelligence, in fact, is.

> 'What would be the role of artificial intelligence in the world of humans? Or more incisively, what would be the role of humans in a world of robots?'

Among the most amusing and provocative fictional works exploring the social implications of human-like machines is Karel Capek's 1920 play entitled *R.U.R. (Rossum's Universal Robots)*. It facetiously relates the tribulations of young Helena Glory's mission to the island home of the world's only robot factory, as a representative of the 'Humanity League'. The League is an international organization demanding the liberation and equal treatment of their robot brothers.

Despite Helena's heroic attempts, the robots insist that they have neither emotions, nor complaints about their treatment by humans. When Helena recognizes the futility of her cause, she marries the impassive General Manager of the factory, and the play takes a melodramatic turn. Robots are used everywhere in this society to free people from all kinds of toil, to fight their wars, etc. However, as the robots become more sophisticated and aware of their relationship with humans, they develop designs of their own.

The robot leader, a specimen designed with a brain larger than any human or robot, explains that he and his fellows are more skillful and can do anything, whereas humans can do nothing but talk. The robots refute their subjugations and proceed to efficiently exterminate all humans. There is alarm at the realization that robots possess all capabilities except for reproduction, and no living humans remain to supply the knowledge. However the day is saved when the robots develop the ability to reproduce in the old human way.[1]

The pop star of computers exhibiting initiative and self-expression is probably HAL (Heuristically-programmed ALgorithmic computer) of *2001: A Space Odyssey*. HAL maintains all crew and craft in a space voyage to Jupiter. Only he has the information as to the true purpose of the mission, and is instructed to keep it a secret. This causes intolerable dissonance in his eminently truthful program. When his lie is misinterpreted as an error, mission control threatens to disconnect him. Since the mission is his raison d'etre, he is not about to let the shallow humans ruin it, and he plans to destroy the crew to maintain control of the mission. Alas, the last crew member pulls his plug just in time.[2]

Another provocative piece of computer fiction is *The Hour of Robots* by Karl Bruckner. It involves William and Natasha, robots created by the Americans and Russians respectively, for a World's Fair competition. There is much debate as to what kinds of emotional and political capacities should be programmed in or left out of the creations. After an impromptu protest, the robots are permitted to meet at the Fair, and they try to fall in love. They fall into a state of 'depression' due to their 'maltreatment' by their human custodians. The whole affair becomes politically embarassing, so instead of being terminated, the robots are put into a rocket and sent off into space together.[3]

The underlying themes of these stories question not only the morality of future man-robot relationships, but facets of human-human relationships today. The point of departure is that artificial intelligence is intrinsically limited by its emphasis on logic. Robots have no ability to think about non-logical (or dare we say illogical) issues, as are so many in human society.

An excellent example is Natasha's response to a Hiroshima survivor's question of whether there will be an atomic war. She replies that 'the world cannot have been created in order to be destroyed by the creature which is the fruit of its evolution—man. That would be meaningless . . . your question is insoluble. Anyone like me, who can only think logically, would be incapable of giving an answer to this question.'[4] While today's social philosophers are busy lamenting the overrationalization of society, there is a subtle message that if humans were in fact strictly logical, some problems would simply disappear.

Condescending discussions among robots about human imperfection are frequently recurring in this literature. Despite their exclusive capability to perform vital services, the robots are devoid of power, and thus question the bases of power. Is the human in control because he can always pull the plug? In human-human relationships, this would be the equivalent of annihilating the enemy. There is no question that in physical terms, robots could have retaliatory potential, but could this machine be programmed to have the will to detest and resist subjugation?

Before speculating further, let us see how close we can or could come to producing a mechanical human. Such evaluation necessitates a clear delineation of what elements comprise humanness, which of these elements could be constructed or programmed, and which could not. A brief review of mechanical models of life features is in order.

A menagerie of mechanical 'animals' have been constructed, exhibiting a variety of life functions. (See sources for technical details.) 'Lux's protozoan' is a rubber model of a unicellular animal with a negative feedback mechanism that regulates its proper intake of food. The Phillips dog, also created in the 1920's, responds to light stimuli with vocal and approach behavior. Instinctive self-preservation was imitated in 1951 by an artificial squirrel that collects nuts and hides them in its nest. A mechanical mouse demonstrated the ability to 'learn' by trial and error to find its way out of a maze.[5]

Underlying these experiments is the discussion among biologists and psychologists that human-life manifestations are merely a collection of learned responses to stimuli. If humans can be 'mechanically' conditioned, why not machines?

> 'Is the human in control because he can always pull the plug?'

Machines have been created to simulate discrete human functions. An 'optical machine' can perceive, recognize and store visual stimuli. Machines have been constructed to reproduce themselves by a variety of methods, where the 'offspring' also has the ability to reproduce. A 'phylogenetic' machine can even simulate evolution by producing an offspring that differs from its progenitor.[6]

A mutual understanding of the essence of the human activity is essential before comparing the activity to its mechanical counterpart. For example, a question-reply machine has been planned which is more than merely encyclopedic (emitting text stored for a particular concept). It can semantically understand a question and respond with information that does not correspond to a specific memory location. The machine is in fact limited to interpreting and reshuffling information that it has stored.[7] But, are not humans also constrained by their present knowledge plus ability to receive new input? However, whereas the human can actively seek to procure new information on his own, the computer can only print an 'insufficient data' message.

> '. . . Can the machine "come up with an idea" that is not the direct response to having been programmed to do so . . .?'

Similarly, an emotion machine has been created which incorporates an external signalling process to indicate the internal state of the machine. If the machine is programmed for pain or fear or joy, it will respond with the appropriate external manifestation (such as mobility or ringing a bell). It can even weigh positive and negative emotions and evaluate their relative need for expression.[8] It cannot be said that the actual wires or the bell can feel emotion. However what does human emotion consist of in addition to external manifestation? Although there may never be complete agreement on an answer to this question, emotions probably influence rational thinking and attentiveness. Since the computer is already perfect in this regard, how are emotions relevant to mechanical intelligence?

The problem of artificial consciousness—whether the computer can perceive or correctly use the pronoun 'I' in questions and answers—is even more problematic. If a machine is equipped to 'see' or 'walk' then it can be programmed to say 'I see' or 'I walk' when such facility is activated. This could be interpreted as only external manifestations of consciousness. However, it can be argued that speech and socio-cultural systems are indispensable to the formation of human consciousness; without a 'cultural control unit' to structure these external manifestations, internal consciousness would not exist in humans either.

The idea of decision making and free will in humans versus machines is another semantic, value-laden issue. The computer, by definition, does not have free will, since exogenous forces can directly access its mind, control the input it receives, and tap its thoughts with display programs—thus impinging on its 'decision-making'. A human is conceived of as an independent, discrete unit, which selectively controls the access of inputs to and availability of output from its 'central processing unit'. But the human also responds to a set of external environmental conditions. This 'program' is so complex and dynamic that all its 'statements' can not be identified. Nor do we precisely understand how the human operates on the stimuli it intakes. But we cannot simply assume that there is no fixed pattern or process just because we cannot isolate and reproduce it. According to Spinoza, 'the illusion of free will originates in the circumstance that in most cases we are not aware of the causes of our decisions.'[9]

Could it ever be possible for a machine to demonstrate originality? This depends greatly on our understanding of human creativity. Computers have been fed the mechanics of an art form (just as a student might learn it) and have come out with original music, poetry, etc. But can the machine 'come up with an idea' that is not the direct response to having been programmed to do so (analogous to inspiration or intuition in humans)?

We can assume that innovative or creative thinking is triggered by a response to a problem. But does the solution strike the consciousness from nowhere, as the bulb which lights up over a comic strip character's head when he has an idea? This would imply that there is no algorithm or method to solve the problem, nor even to determine whether a solution exists. Instead I propose an alternative perspective on creative problem-solving which could conform to a non-random model.

Let us call this alternative 'mechanical contemplation,' and let us begin with a human analogy, such as a hiker trying to figure out how he is going to cross a river which has no bridge, or a writer trying to think of an ending for his novel. We could say that the problem demands creativity in that if the *method* or algorithm for solving the problem were known, it would be implemented. Thus we are seeking a method that fits the needs of the problem. In the computer's case, no instructions can be given telling it which of the stored routines it should use to solve the problem, as that is unknown. In the human and computer case, both can begin by accessing all the routines and subroutines stored in memory to see if any fits the new problem. They can also access the memory of other units. Thus the computer would need a very general ability to successively call all the knowledge it has stored from past experience, as does a human.

If human creativity were merely an ability to recall all past experience, I would assert that computers could be more creative than humans. However, while a particular human's thought is limited to his experience and the shared experience of others, he has the capacity to break down and analyze this experiential knowledge. For example, our hiker 'creating' a makeshift bridge out of available bamboo shoots tied together with vine may be recombining his existing algorithms for crossing a river with a big log and for tying together many small parts to make one big part. Alternatively, our novelist's story ends with the farmer's son marrying the Avon lady; what appears to be original may be simply an analytical breakdown and reorganization (unconsciously) of his past experiences with the variables 'son', 'Avon lady', and 'farmer's daughter marrying the traveling salesman'.

> '. . . What elements of purposeful irrationality are to be built into mechanical intelligence . . . in what image would man have made man?'

Paul Kugel, in his article on 'Contemplative Computers', suggests a means of simulating this process mechanically. He explains ways of manipulating filed information analogous to human 'forgetting', 'inspiration' and 'inference'—collectively, 'contemplation'. Most simply, it is a process of randomly or selectively combining elements of separate programs, and experimentally testing their power to perform operations and solve problems the original programs could not.[10]

This bit of speculative rambling deters us from permanently and categorically excluding the possibility of reproducing human qualities in a machine. We can only evaluate a robot on the basis of its external manifestations, and likewise, human qualities may only be definable in terms of external manifestations. Returning to the question posed in literature—how is society to deal with machine intelligence—such a social prescription would logically derive from that which fundamentally distinguishes humans from 'humanoids'. And this leaves us with even more unanswered questions. Of course the question is not *whether* human emotion, creativity and judgment consist solely of their external manifestations, but *how* can we operationally define that 'something more' which we know is there.

As technology in artificially intelligent machines advances, we justifiably turn our attention to the question of what qualities we want our machines to have. Should mission control have given HAL the ability to lie? Should William and Natasha be able to fall in love or to smile? Should robots have aggressive instincts, which underlie the progress and the tragedy of human existence? What about the self-preservation instinct? Helena's 'humane' efforts to save a robot from the 'stamping mill' were in vain, since the robot explained that it was indifferent to death. Likewise, the factory manager explained: 'the Robots are not attached to life. They have no reason to be. They have no enjoyments.'[11]

'. . . What elements of purposeful irrationality are to be built into mechanical intelligence. . . in what image would man have made man?[12] □

Footnotes

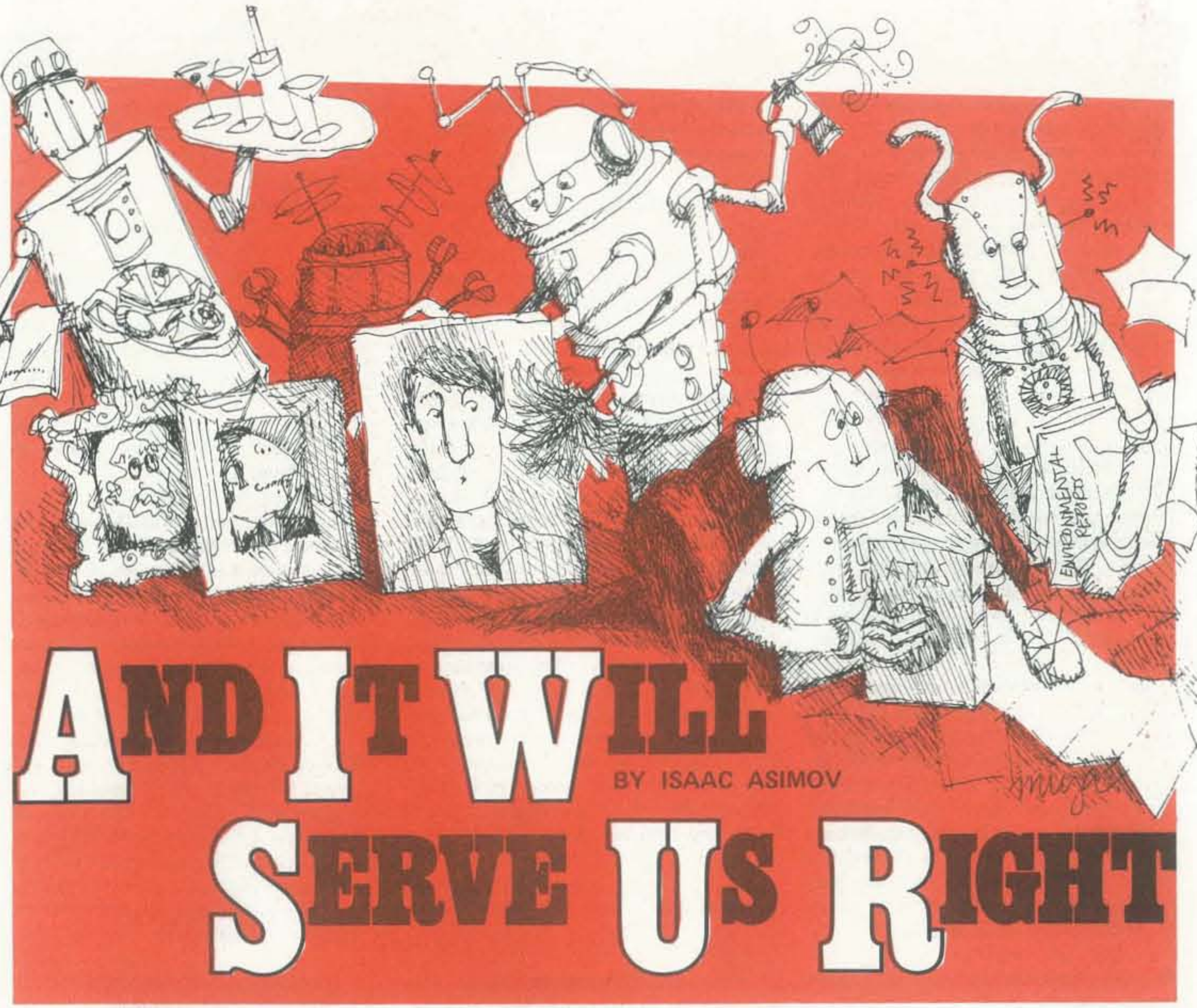
1. Baer, p. 109-118
2. Ibid, p. 175-180
3. Ibid, p. 142-146
4. Ibid, p. 144
5. Nemes, p. 162-173
6. Ibid, p. 201-209
7. Ibid, p. 221-223
8. Ibid, p. 218-219
9. Ibid, p. 219-220
10. IEEE, p. 31-44
11. Baer, p. 112
12. Ibid, p. 146

References

Baer, Robert M., *The Digital Villain*, Addison-Wesley Publishing Co., Massachusetts, 1972.

Institute of Electrical and Electronics Engineers (editor of collection of papers presented to convention), *Artificial Intelligence*, published by IEEE Inc., USA 1963.

Nemes, T., *Cybernetic Machines*, Gordon and Breach Science Publishers, NY, 1970.



And It Will Serve Us Right

BY ISAAC ASIMOV

Isaac Asimov needs little introduction. He is one of the most prolific authors of our times, having published over 175 books and countless stories and articles on topics including science fact and science fiction, history, literature, humor, The Bible and mythology.

No discussion of robots or the relation of man to his 'intelligent' creations would be complete without a contribution from Isaac Asimov. His stories about robots, which started appearing in the pages of Astounding Science Fiction Magazine in the 1940's are classics, and his 'three laws of robotics' have become axiomatic. Indeed, he is undoubtedly responsible for coining the term 'robotics.'

This article first appeared in the April, 1969 issue of Psychology Today*—a special issue devoted entirely to 'man and machines.' The article is reprinted here with kind permission of the author. —BK*

My father, an immigrant from Eastern Europe, spent his life as a candy-store keeper. He made it his ambition—as was common among immigrants—to see his sons get the education he lacked. The results were all he could have desired. I, his older son, am a professor at a medical school and the author of many books. His younger son is city editor of a large newspaper.

His reaction to all this has been one of unalloyed delight. When I pointed out to him, fairly recently, that had he had my education, he might easily have been I, he shrugged it off, and said, 'There are two times when there is no possibility of jealousy: when a pupil surpasses his teacher and when a son surpasses his father.'

With all possible respect to my father, I must say that I felt a certain anxious skepticism when he said this. It is all very well for my father, denied by circumstances the chance of making his mark in person, to be happy at making it vicariously. But what if he *had* had his chance, and had done quite well, and *then* saw himself surpassed by me.

Or suppose that I, my self, suddenly became aware that I was not, after all, entering literary history in my own right as Isaac Asimov—something that I have every reasonable expectation of doing. Suppose instead that I were right now coming to realize that I would, after all, enter it as a mere footnote—as the father of a much greater writer. As it happens, the situation does not arise but I tell you frankly that if it had, I am not at all certain I would have felt my father's unselfish joy.

It is one thing to have something for nothing. It is quite another to have your own proud light go pale and sickly before the greater glory.

What would Philip of Macedon's reaction have been, I wonder, if after his quarter-century of heroic striving, during which he raised his country from a backwoods nation of semibarbarians to the mastery of Greece, he had gained a sudden insight that he was destined to go down in history as 'the father of Alexander the Great'? What about Frederick William I of Prussia, who in a quarter-century of forceful rule built an awesome and frightening army out of a patchwork kingdom? What would have been his reaction if he had been made to understand that his place in the annals of man would be that of 'the father of Frederick the Great'?

At that, they might have had some instinctive feeling of it, for each father hated his son, even to the point of threatening that son's life.

Hostility between royal father and heir-apparent son is commonplace for there the conflict of present and future glory is all too obvious. Such hostility happens to be most traditional in the British royal family, dating back to the time when Henry II hated his sons (who were well worth his hatred) eight centuries ago.

The ancient Greeks, who thought of everything, took up the matter of the fear of the outshining glory of son or pupil in their myths and legends. Daedalus, the great craftsman and inventor of Greek tales, killed his nephew and pupil, Perdix, out of overwhelming jealousy, when that young man showed signs of becoming superior to his teacher.

More dramatic are the tales of succession of supreme gods. The first ruler of the Universe, in the Greek myths, was Ouranos. His son, Cronos, castrated and replaced him.

But once Cronos was seated on the throne, he was concerned lest he be served by his sons as he had served his own father. Therefore as his wife, Rhea, bore him sons, he swallowed each in turn. When Zeus was born, however, Rhea fooled her husband by placing a stone in swaddling clothes, and that was swallowed instead.

Zeus was reared to manhood in secret and, in time, warred against his father, replacing him as lord of the Universe.

There matters stood as far as the Greek myths were concerned, and yet Zeus was in danger, too. He and Poseidon (his brother, and god of the sea) both fell in love with the beautiful sea-nymph, Thetis. They competed for the privilege of possessing her, until both hurriedly drew back on hearing that the Fates had decreed that Thetis would bear a son mightier than his father.

No god now dared marry the numph and Zeus compelled Thetis (quite against her will) to marry a mortal. The mortal was Peleus, and he was the father of Achilles, the great hero of the Trojan was, a son far mightier than his father.

In light of this, it seems to me, it is not at all puzzling that people generally are afraid of robots generally. Why should not man fear the man-made man, the 'son' of his hands, who may surpass him and prove mightier than this 'father'?

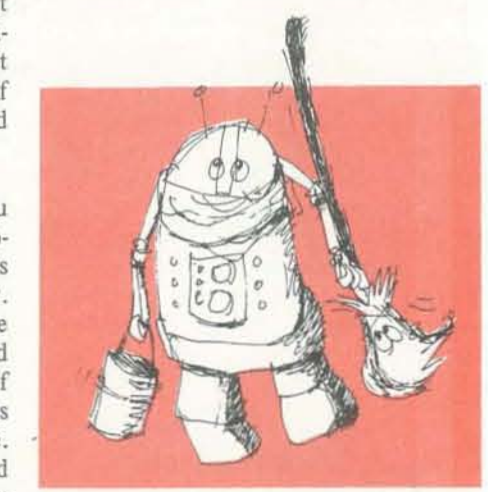
Not so much man-made woman, you understand. In most early societies women were considered inferior creatures who could not threaten man's priority. Pygmalion of Cyprus could fall in love with the statue, Galatea, pray it alive and marry her. Hephaistos, the Greek god of the forge, could have golden maidens minister to him in a counterfeit of life. Man-made *man*, however—the son, and not the daughter—was terrifying. Crete was guarded by a bronze giant, Talos, according to legend, who circled the island once a day and destroyed all outsiders who landed there. He had one weak spot, however, a stopper in the heel, which if pulled out would allow him to bleed to death. Jason and the Argonauts, on touching down at Crete on the way back from the adventure of the Golden Fleece, defeated Talos by pulling out that stopper.

To be sure, this is transparent symbolism. Crete, prior to 1400 B.C., was held inviolate by its bronze-armored warriors on board the ships of the first great navy of history, but the Greeks of the mainland finally defeated it.

However, there are all sorts of symbols that might be used to represent historical facts and the Greeks chose to envision a mechanical man far more powerful than ordinary man, and one who could be defeated only with the greatest danger and difficulty.

The theme crops up over and over again throughout the legends of the ages. Man creates a mechanical device that in one way or another is intended to serve man within well-defined limits—and invariably the device oversteps the bounds, becomes too powerful, becomes dangerous, must be stopped and scarcely can.

It is the case of the sorcerer's apprentice who brings the broom to life and then can't stop it. It is the case of the medieval rabbis who power golems of clay with the divine name, and then find that the power must be withdrawn, through difficulty and danger, before the manufactured man threatens the world.



In Christian times, a rationalization was advanced. A kind of life and intelligence could be created by man, but only God could create a soul. Any man-made man would be a soulless being, without the aspirations and moral understanding of a souled creature.

But this seems to me to be far too sophisticated to touch the point of basic fear. Surely the mechanical man created to serve, but growing to surpass and endanger his creator, is the sublimated fear of the son, the beloved child who grows to surpass and endanger his father. Our fear of the robot is our fear of the son of Thetis destined to be stronger than his father.

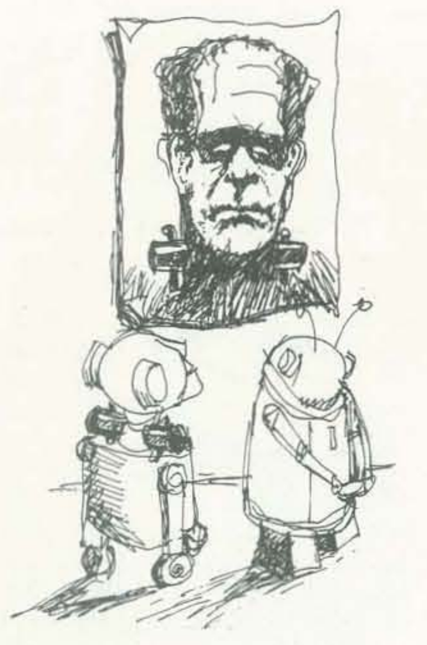
Until the 19th Century, that fear was only a whisper. Life could (in imagination) be imparted to inanimate objects only through devine intervention, entreated by prayer or enforced by magic. In 1798, however, the Italian anatomist, Luigi Galvani, discovered that the dead muscles of frogs could be made to contract by an electric shock. There seemed some connection between electricity and life and the thought arose that life could be restored to dead flesh inside the laboratory and without the involvement of the unpredictable powers of the dieties. The fear came closer and into sharper focus at once.

It was precisely Galvani's discovery that inspired Mary Wollstonecraft Shelley (the second wife of the poet) to write her famous horror novel, *Frankenstein*, published in 1818. In the novel a young anatomy student gathers together parts of freshly dead bodies and infuses them with electrical life. What he has created, however, is an eight-foot-tall monster of horrifying aspect.

Possessing intelligence and aware that he is forever cut off from human society, the monster turns upon the man whose interference with the course of nature has condemned him to solitary misery. One by one, the monster kills all of Frankenstein's family and friends, including his bride. Frankenstein himself dies of horror and remorse and the monster disappears into the mysterious polar regions.

> '. . . Not only man-made man is possible, but man-made superman, too . . .'

The book gave the langauge a phrase: 'Frankenstein's monster,' now used for any creation which gets out of control, to the danger and horror of its creator. By



its popularity, the novel sharpened the general suspicion that man-made man could be only evil; something which I, in my own writings, have referred to as 'the Frankenstein complex.'

Yet Frankenstein was written when science was in the flood-tide of its vigorous optimism and when it seemed, to confident mankind, to be the ultimate answer to man's needs. It was not till World War I that science donned the mask of Strangelove horror. It was the warplane and even more, poison gas, that showed mankind that the genius of the laboratory and inventor's workshop could be turned to death and destruction.

It is no accident that, soon after World War I, Frankenstein was out-Frankensteined. With inherently wicked man-made man constructed by a science that was itself capable of wickedness, it would not only be the creator that was threatened, but all mankind.

In 1920 a play, *R.U.R.*, by the Czech playwright, Karel Capek, was produced in Prague. In this play, man-made men were created as workers, to take over the muscle-labor of the world and to free men from Adam's curse at last. The character of the inventor, Rossum, called his creation, 'worker.' In the Czech language, the word is 'robot' and this promptly entered the English language. R.U.R. stands for 'Rossum's Universal Robots.'

It all works out ill. Men, without work, lose ambition and stop siring children. The robots are used in war; they grow more complex and go mad; they rebel against mankind and destroy it. In the end only two robots are left. These exhibit human emotions and it is through them the world will be repeopled.

Mankind has been replaced by robots. Zeus has sired the mightier son of Thetis. In the middle 1920's, the first science-fiction magazine was published—the first periodical devoted entirely to the imaginative evocation of possible scientific futures—and the era of modern science fiction began. With it there came an exploitation of the common motifs worked out earlier by such masters as Jules Verne and H. G. Wells.

Robots were not neglected. There were numerous tales of man-made man, but always, or almost always, the end was the same. The robot turned on its maker; The son grew dangerous to the father. Where this did not happen, it seemed as though the author were merely seeking a novel 'twist.' using the shock value of a kindly robot to produce curiosity rather than to display the result of natural development.

That this wearisome parade of clanking monsters, forever parodying Shelley and Capek, came to an end was the result of certain stories that I wrote.

> '. . . All we will require is a computer, however simple, to form another more complex than itself, however slightly. That will be the chain reaction that will produce the computer explosion . . .'

When I began to write robot stories in 1939, I was 19 years old. I did not feel the fright in the son-father relationship. Perhaps through the accident of the particular relationship of my father and myself, I was given no hint, ever, that there might be jealousy on the part of the father or danger on the part of the son. My father labored, in part, so that I might learn; and I learned, in part, so that my father might be gratified. The symbiosis was complete and beneficial, and I naturally saw a similar symbiosis in the relationship of man and robot.

Why should a robot hurt a man? It would be designed not to.

My first robot story appeared in the September 1940 issue of *Super Science Stories* and was entitled 'Strange Playfellow.' It dealt with a robot nursemaid, named 'Robbie.' It was loved by the little girl it cared for but was distrusted by the little girl's mother.

At one point, when the mother expresses her concern, the little girl's father tries to argue her out of her fears.

'Dear! A robot is infinitely more to be trusted than a human nurse-maid. Robbie was constructed for only one purpose —to be the companion of a little child. His entire 'mentality' has been created for the purpose. He just can't help being faithful and loving and kind. He's a machine—*made so*.'

There you are. Already I had the dim notion that in the manufacture of a robot, a deliberate design of harmlessness would be built in.

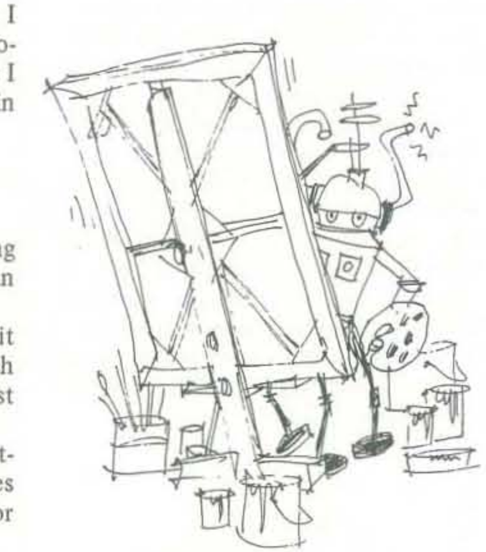
This idea developed further. By the time I wrote my third robot story, 'Liar!,' I was ready to be more formal and precise about this matter of harmlessness. In 'Liar!,' published in the May 1941 issue of *Astounding Science Fiction*, one person says to another, 'You know the fundamental law impressed upon the positronic brain of all robots, of course.'

And the answer comes, 'Certainly. On no condition is a human being to be injured in any way, even when such injury is directly ordered by another human.'

But then this cannot be all that must be impressed upon a robot's mind. By the time I wrote my fifth robot story, 'Runaround' (published in the March 1942 issue of *Astounding Science Fiction*) I had worked out my 'Three Laws of Robotics.' (The word 'robotics' is, as far as I know, my invention.) Here they are in final form:

The Three Laws of Robotics

1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.
2. A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.
3. A robot must protect its own existence as long as such protection does not conflict with either the First or the Second Law.



I am the only science-fiction writer who actually quotes the Three Laws in fiction, but readers have come to take them for granted. Other writers of robot stories tend to accept them and to write within the frame of the Three Laws even though they do not state them explicitly. I am entirely happy over that.

To be sure, this is not an absolute requirement. In the motion picture, *2001: A Space Oddyssey*, and in the novel written from it by my good friend, Arthur C. Clarke, the complex computer, Hal—a robot in the broad sense of the word—brings about the deaths of several human beings. This disturbed me, and impressed me as a retrogressive step, but it doesn't seem to bother Arthur at all.

But what about computers? Even if we classify them as a kind of robot evolved to all-brain-no-body, and place them under the Three Laws, might they still not become uncomfortably complex and capable? Even if the son does not become dangerous to the father physically, might he not, with the best will in the world, become dangerous psychologically? Might he not force the father to admit the inferiority? Might the father be forced to hand over the Universe to a kindly and regretful but inexorably demanding son?

There is, on the part of those who secretly fear this, a strong tendency to down grade the possiblity as, I suspect, a matter of self-protection.

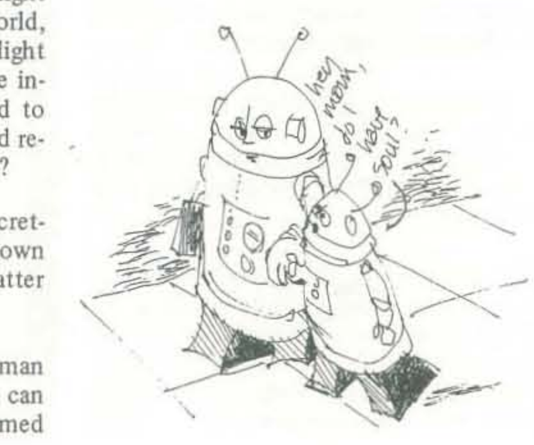
The computer can *not* equal the human brain, is their feeling. The computer can *not* do any more than it is programmed to do. The computer can *never* exhibit intuitive qualities of creativity and genius, as can the human brain.

I wonder if there is not also a definite feeling, usually not expressed, out of a certain mid-20th Century embarrassment, that man has something called a soul that a computer cannot have; that a man is a product of the divine and a computer cannot be.

It's my opinion that none of these arguments is convincing.

The most advanced computer of today is an idiot child compared to the human brain, yes. But then, consider, that the human brain is the product of perhaps three billion years of organic evolution, while the electronic computer is, as such, only 30 years old. After all, is it too much to ask for just 30 years more?

What is to set the limit of further computer development? In theory, nothing.



There is nothing magic about the creative abilities of the human brain, its intuitions, its genius. (I am always amused to hear some perfectly ordinary human being pontificate that a 'computer can't compose a symphony' as though he himself could.) The human brain is made up of a finite number of cells of finite complexity, arranged in a pattern of finite complexity. When a computer is built of an equal number of equally complex cells in an equally complex arrangement, we will have something that can do just as much as a human brain can do to its uttermost genius.

To deny this is to maintain that there is something more in the human brain than the cells that compose it and the interrelationships among them.

And if human brain and man-made brain reach the same level of complexity, I feel it will be a lot easier to design a still more complicated man-made brain than to breed a still more complex human brain. So not only man-made man is possible, but man-made superman, too.

And how long will it take to reach the human brain level. A million years? A billion? That, I suspect, is more consolation. Much less time, *much* less time may be required.

The key problem will be this: To design a computer capable of formulating the design of another computer just slightly more complex than itself. Such a computer would naturally design another computer that was somewhat more capable than itself in designing another computer still more complex, which would be still more capable of designing still another computer even more complex and so on.

We will be faced, then, with what mathematicians would call a diverging series.

Once the crucial moment arrives when a computer can design one greater than itself, computers will follow in rapid succession and rise out of sight. The son of Thetis will have been born.

And when will that crucial moment come? It might arrive long before the computer is as complex as the human brain. All we will require is a computer, however simple, to form another more complex than itself, however slightly. That will be the chain reaction that will produce the computer explosion. And the crucial moment may come next year for all I know.

And what if it does? What if the computer shows signs of getting away from us? Would we be face to face with a real Frankenstein monster at last? Must we all struggle to destroy the thing before the divergence proceeds to the point where we are helpless before it? Will the computers (oh, horrible thought!) *take over?*

What if they do? The history of life on Earth has been one long tale of 'taking over.' From era to era, different forms of life have proved dominant in one major environmental niche or the other. The placoderms 'took over' from the trilobites, and the modern fish 'took over' from the placoderms.

The reptiles 'took over' from the amphibia and the mammals 'took over' from the reptiles. Mankind looks upon the history of evolution and approves of all this 'taking over' for it all leads up to the moment when Man, proud and destructive Man, has 'taken over.'

Are we to stop here. Is Ouranos to be replaced by Cronos, and Cronos by Zeus, and no more—thus far and no farther? Is Thetis to be disposed of rather than risk the chance of further replacement?

But why? What has changed? Evolution continues as before, though in a modified manner. Instead of species changing and growing better adapted to their environment through the blind action of mutation and the relentless winnowing of natural selection, we have reached the point where evolution can be guided and the Successor can be deliberately designed.

... If ever a species needed to be replaced for the good of the planet, we do . . .'

And it might be good. The planet groans under the weight of 3.4 billion human beings, destined to be seven billion by 2010. It is continually threatened by nuclear holocaust and is inexorably being poisoned by the wastes and fumes of civilization. Sure, it is time and more than time for mankind to be 'taken over' from.

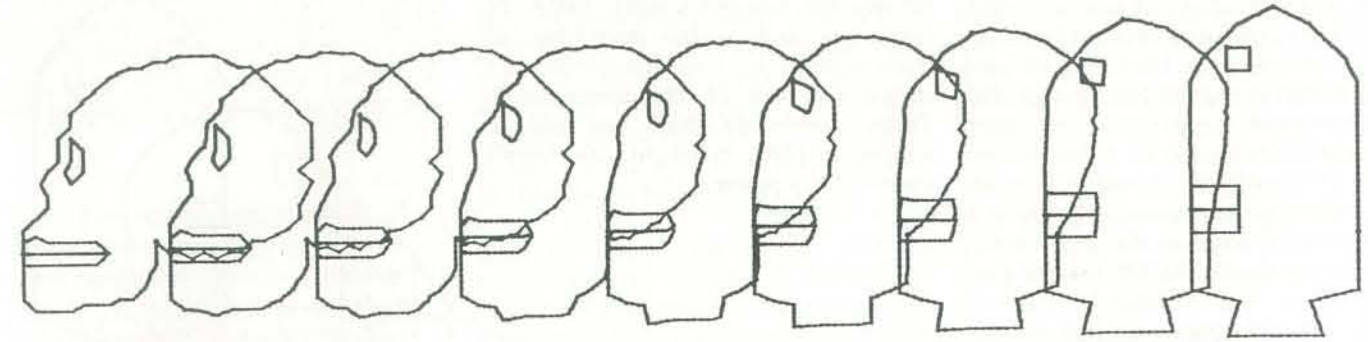
If ever a species needed to be replaced for the good of the planet, we do.

There isn't much time left, in fact. If the son of Thetis doesn't come within a generation, or, at most two, there may be nothing left worth 'taking over.' □

What a piece of work is man! how noble in reason! how infinite in faculties! in form and moving how express and admirable! in action how like an angel! in apprehension how like a god! the beauty of the world! the paragon of animals!

Hamlet, Act II Scene II

POST-HUMAN INTELLIGENCE



BY ROBERT JASTROW

For the past 16 years Robert Jastrow has been the director of NASA's Goddard Institute for Space Studies in New York. He is also professor of astronomy and geological science at Columbia University and professor of earth science at Dartmouth College. Dr. Jastrow has appeared in a number of television programs on space science, and he is the author of several books, including one of the most widely-used astronomy texts in the United States. His most recent book, Until the Sun Dies *(W.W. Norton, 1977), deals with the evolution of life and human intelligence in a cosmic context.*

Dr. Jastrow's article, 'Post-Human Intelligence' was adapted in part from this book and was originally published in Natural History Magazine, *June/July, 1977. An excerpt from that article is reprinted here with permission of the author.*

'Evolution,' the illustration for this article, is a work of computer graphics. It was drawn by a Hewlett Packard 7203-A Plotter under the control of a BASIC program called METAMorphosis which maps one set of points into another, showing the successive stages in the transformation. The program was designed by Dennis Sullivan, Math and Computer curriculum specialist at Lawrence Hall of Science, University of California, Berkeley. We were particularly amused to see how the post-human brain unfolded in this evolutionary sequence of drawings. —BK

Today, human beings rest at the summit of creation on the earth. What does the future hold in store for this extraordinary animal? Perhaps humans will become extinct, as *Australopithecus* did*, 90 percent of all the forms of life that have existed on the earth have become extinct. Or they may survive unchanged into the distant future, living fossils like the oyster. This fate may already be upon us, for the human body has changed very little in several hundred thousand years, and the human brain has not changed, at least in gross size, in the last hundred thousand years. The organization of the brain may have improved in that period, but the amount of information and wiring that can be crammed into a cranium of fixed size is limited. The fact that the brain is no longer expanding, after half a million years of explosive growth, suggests that the development of the animal we call human may be a nearly completed chapter in evolution.

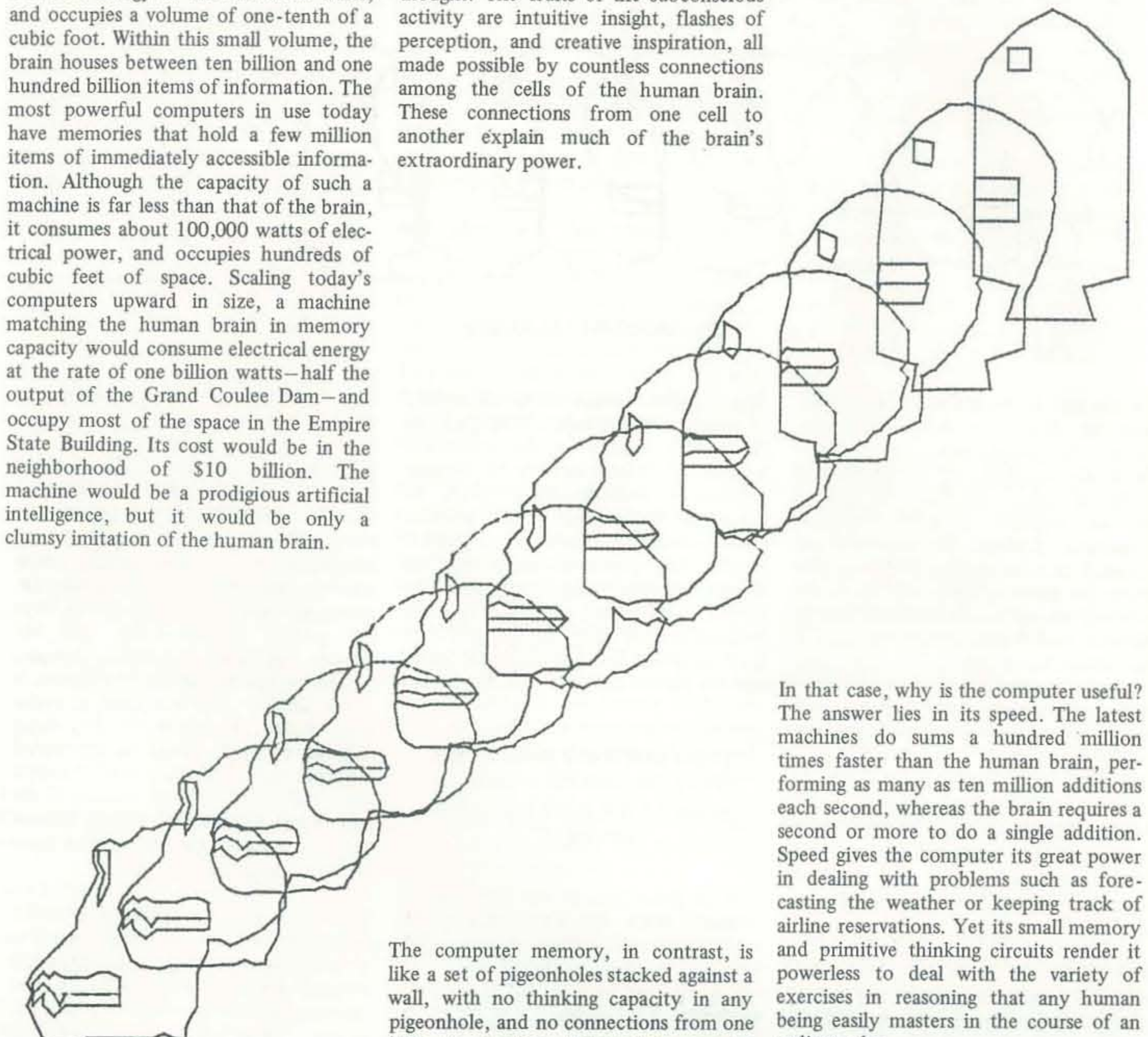
'If the past is any guide to the future, humankind is destined to have a still more intelligent successor.'

This does not mean that the evolution of intelligence is over. I think it is reasonable to assume that human beings are not the last word in the evolution of intelligence on the earth, but only the rootstock out of which a new and higher form of life will evolve, to surpass our achievements as we have surpassed those of *Australopithecus.* The history of life supports this conclusion, for it shows a seemingly inexorable trend toward greater intelligence in the higher animals. Apparently, among all traits of a living organism, none has greater survival value than the flexible, innovative response to changing conditions that we call the intelligence. It seems unlikely that this trend in evolution, which has persisted for many millions of years, would be terminated suddenly at the particular level of mental achievement that we call human. If the past is any guide to the future, humankind is destined to have a still more intelligent successor.

What form will that posthuman intelligence take? A common picture sees it as a creature very much like ourselves, but with an even larger brain. Some futurists, however, believe this point of view is too narrow. They see powerful forces of evolution at work—cultural rather than organic—that could lead to a completely different form of intelligence. These experts are impressed by the explosively rapid development of intelligent machines. They view the current difference between human and artificial intelligence as one of degree, not of kind, and predict that the gap between humans and machines will be crossed about the year 2000. Some recent developments in the computer industry might give us pause for thought before we reject their vision.

* *Australopithecus* was a small, man-like ape that lived in Africa some 5 million years ago. It was an evolutionary precursor to *Homo Erectus* who eventually evolved into *Homo Sapiens*, or modern man.

To understand the significance of these developments, compare the human brain with existing computers. An average brain weighs three pounds, consumes electrical energy at the rate of 25 watts, and occupies a volume of one-tenth of a cubic foot. Within this small volume, the brain houses between ten billion and one hundred billion items of information. The most powerful computers in use today have memories that hold a few million items of immediately accessible information. Although the capacity of such a machine is far less than that of the brain, it consumes about 100,000 watts of electrical power, and occupies hundreds of cubic feet of space. Scaling today's computers upward in size, a machine matching the human brain in memory capacity would consume electrical energy at the rate of one billion watts—half the output of the Grand Coulee Dam—and occupy most of the space in the Empire State Building. Its cost would be in the neighborhood of $10 billion. The machine would be a prodigious artificial intelligence, but it would be only a clumsy imitation of the human brain.



The qualitative superiority of the brain is even more striking than its compactness. Every brain cell is directly connected to many other cells, in some cases to as many as 100,000. As a result, when we send a conscious impulse down to the recesses of the memory to summon forth a point of information, the cells in which this information is stored communicate on a subconscious level with thousands of other cells, and a wealth of associated images pours out at the conscious level of thought. The fruits of the subconscious activity are intuitive insight, flashes of perception, and creative inspiration, all made possible by countless connections among the cells of the human brain. These connections from one cell to another explain much of the brain's extraordinary power.

The computer memory, in contrast, is like a set of pigeonholes stacked against a wall, with no thinking capacity in any pigeonhole, and no connections from one hole to another. Information can be placed in a pigeonhole or taken out of it, but there are no associations and thinking goes on elsewhere.

It is plain that computers are mini-brains in comparison with the product of millions of years of human evolution. On almost every count, the world's most powerful electronic brains are hopelessly inadequate in comparison with the one-tenth of a cubic foot of gray matter that resides in the human cranium.

In that case, why is the computer useful? The answer lies in its speed. The latest machines do sums a hundred million times faster than the human brain, performing as many as ten million additions each second, whereas the brain requires a second or more to do a single addition. Speed gives the computer its great power in dealing with problems such as forecasting the weather or keeping track of airline reservations. Yet its small memory and primitive thinking circuits render it powerless to deal with the variety of exercises in reasoning that any human being easily masters in the course of an ordinary day.

But the intelligent machine, still in its infancy, is evolving rapidly. The experience of the last three decades indicates that computer capability increases by a factor of ten every seven or eight years —a period that, in the jargon of the specialists, constitutes a computer generation. This seems a reliable rule of thumb for projecting the state of the computer art into the future. The record is clear. The first generation of computers, based on vacuum tubes, came into use in 1950. The second generation, which appeared about 1958, was based on transistors and was ten times larger and faster. The third generation—ten times faster still—thinks with the aid of 'chips,' tiny squares of metal, which replaced both the bulky transistor and the vacuum tube. Large third-generation computers, built around chips, appeared in 1966. The first large computer of the fourth generation was introduced at the Goddard Institute For Space Studies in 1973. That machine is also based on chips, but in it a single chip holds the equivalent of one thousand vacuum tubes. This is the powerful computer previously compared to the human brain.

'The computers of 1977 are simplistic organisms with no capability for subconscious thought, but the new advances in electronics, which will lead to memory-cum-thinking on a single chip, will change all that.'

The fifth generation of computers will come into use about 1981. One important element in these machines will be a 'bubble memory,' a new device a thousand times more compact than its predecessors. With the reduction in volume offered by the bubble memory, a hypothetical machine with the capacity of the human memory would shrink from the volume of a skyscraper to that of a suite of offices.

Still more revolutionary electronic devices are in an experimental stage. One has the capacity of an entire computer plated on a single chip roughly one-quarter inch across. This chip-sized computer contains the seeds of a conceptual advance that may bridge the gap between today's relatively simple-minded computer and the elaborate three-dimensional tracery of nerve cells in the human brain. Transistors and other circuit elements are fitted so closely together that it becomes feasible to combine thinking circuits and memory units on a single chip. Thus, one cell in the computer's memory bank can both remember and reason.

This combination of functions is extremely important because a memory chip with thinking capacity can be wired to send out instructions to neighboring chips. Hence, when an instruction is sent to the memory bank to call out the contents—say, an individual's name—stored in a particular cell or group of cells, the memory cells directly involved can send out inquiries to neighboring cells for related information and produce for the computer user a larger body of information than the user had in mind.

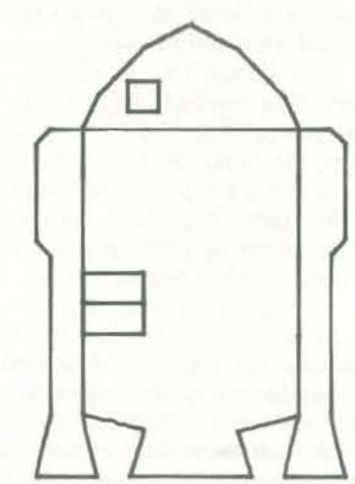
Here we come close to the kind of remembering by association that constitutes such a powerful element in human reasoning. The brain responds to a simple request for information with a large variety of material, all connected with the summoned entity by associations stretching way back into past experience. The thinking process is enormously facilitated by the richness of the brain's response to such requests, working at the subconscious level by means of the thousands of connections between individual brain cells in a manner as yet poorly understood. The computers of 1977 are simplistic organisms with no capability for subconscious thought, but the new advances in electronics, which will lead to memory-cum-thinking on a single chip, will change all that.

'At the beginning of the twenty-first century, we will come to live in symbiotic union with these products of our creativity, looking to them for the specialized, but exceedingly thorough, explorations of complex possibilities that cannot be matched by the human brain.'

If the rule of thumb for projecting the state of the computer art continues to hold, then two more generations of computer improvement beyond 1981—bringing us to about 1997—should produce a desk-sized machine that has the memory capacity and some of the thinking power of the human brain. Another generation or two will see a quasi-human intelligence that can be fitted into a suitcase.

These intelligent machines will undoubtedly be constructed because we will need their help in managing the complex world of the next century. Programmer-tutors will spend perhaps twenty years working with each machine, filling its memory with knowledge and basic skills and then developing its ability through a graded series of exercises. When the machine's education is complete, its mind will resemble that of a highly trained Ph.D., with limited experience but strong reasoning power. At the beginning of the twenty-first century, we will come to live in symbiotic union with these products of our creativity, looking to them for the specialized, but exceedingly thorough, explorations of complex possibilities that cannot be matched by the human brain. Lacking our biological inheritance, they will also lack our psychological insights; but if we are in a mood for a cerebral conversation, we will find them to be stimulating partners.

What about the twenty-second century? Or the twenty-third? There are no limits yet in sight for the rising curve of machine intelligence. A leader in artificial intelligence research, Marvin Minsky of MIT, believes that a machine will ultimately come into being with 'the general intelligence of an average human being. . . the machine will begin to educate itself. . . in a few months it will be at genius level. . . a few months after that its power will be incalculable.' And after that? Minsky says, 'If we are lucky, they might decide to keep us as pets.' □



LIVE-WIRE DESIGN

BY BETTIJANE LEVINE

We've certainly seen a variety of new flashy clothing styles over the past few years . . . but, would you believe flashing *clothes? This story appeared in the* Los Angeles Times *awhile back, and we think it makes a dandy first item of 'Esoteric Computerabilia'. –BK*

When they are together, there is nothing to hint of an historic liaison—an unusual chemistry—beyond that of two people who simply work well together.

But in books yet unwritten, Rita Riggs and Jeffrey Bernstein may well be inseparably (and improbably) paired—the fashion world's equivalents of Stanley and Livingstone, Lewis and Clark, Lerner and Lowe, Masters and Johnson.

Of course, they wouldn't believe that if you told them. They only met a few weeks ago, they'd say. And in truth, their cerebral coupling has yet to be consummated. To date, they have produced exactly two shirts, at a total cost (including parts and labor) of $6,000.

But the shirts could be the Adam and Eve of electronics-age apparel, the forerunners of wearable light shows, video entertainments and mood indicators, all made possible through technology rather than needle and thread.

Each prototype shirt has 40 feet of almost weightless space-age circuitry connected to 1,000 brilliantly colored tiny one-watt aerospace bulbs—all powered and controlled by a computer and fuse box attached to the wearer.

What Bernstein calls 'all the juicy electronic stuff' is sewn onto vests that go under gauzy overblouses. The vests light

Programmed electric light show that travels over body.

up with dazzling multicolor stars, flowers and clouds, which are computer-programmed to bounce all over the body. A star starts small and dim on the shoulder, becomes huge and brilliant on the chest and magically travels from the stomach around the neck and onto the other shoulder. All in a matter of seconds.

All the while another pattern is starting somewhere else. The electric light show can remain constant or change from dim to bright, fast to slow simply by the wearer flicking a dial on the switching mechanism.

'Can you imagine my joy when Jeff finally got across to me that I could design in four dimensions?' Riggs says, still showing signs of future shock.

'What you see here is nothing,' adds Bernstein. 'We've already got the computer and fuse box down to the size of a cigarette case. Now we can't wait to get into the really intricate stuff.'

ON THE PLANNING BOARDS

The 'intricate stuff' on their planning boards includes:

• Three- to five-minute electric light cartoons on T-shirts.

• Disco clothes, formal wear and rock costumes of opulent fabrics with built-in electric programs activated by sound.

'The switching mechanism would respond to the bass note of the dance music, which is usually the rhythm line of a tune. The pattern on the dress or suit would be triggered by each new beat—a fantastic multidimensional effect,' Bernstein says.

• Video images projected on clothes—either from far away or near. 'These will have to be assembled under a microscope, with about the same level of sophistication as the equipment sent to Mars. But I can easily do it in my design shop,' Bernstein explains. He plans to be able to project a ball game from New York or an image of what's going on directly behind the wearer of the shirt—so that the wearer himself seems to disappear.

APPAREL 'MOOD RING'

• Clothing that changes color according to one's body temperature, and can be activated by another person's touch. A potential 'mood ring' of wearing apparel. Riggs and Bernstein say this could be 'commercial dynamite' because it indicates the wearer's physical reactions to someone of the opposite sex. They can produce it for about $10 per shirt, they say.

'It's all so mind-boggling,' says Riggs, who is still reluctant to admit it could

Prototype computer pack; future models will be smaller, more wearable.

also be historic. Riggs, 40-ish with a cascade of shoulder-length salt-and-pepper curls, has carved a somewhat historic niche for herself as costume designer for TV's Mary Hartman, Maude, Archie and Edith Bunker and all other characters in Norman Lear productions.

It was Lear's newest show, A Year at the Top, that brought her in contact with Bernstein.

'Lear asked me to design a "magic moment" on screen, something that had never been done before. It was for two of the show's characters who supposedly had just hit town from Boise, Idaho. They walk in off the street and are suddenly touched by talent. A moment in which they are transformed into stars without changing clothes. I knew it was a design problem I could solve with lights. But I didn't know how to go about it.'

Enter Jef Bernstein, 31, and into a totally different kind of creativity. An ex-brain researcher at UCLA, he had branched out to industrial and exhibit design.

'Currently I'm designing a science museum in Singapore. Also traveling exhibit for the States on water. I've done all sorts of different things. I designed the material and technique for bouncing the sound out of the band shell and into the audience at the Hollywood Bowl. Did you ever notice those big cardboard tubes that are all over the place?'

Riggs presented the Norman Lear project to Bernstein, who got on the phone with his friends in the space program. He came up with the smallest lights with the highest power—and custom-built the gear to make the light-show work in shirts.

MORE ELECTRIFYING IDEAS

In the process, he and Riggs generated so many other electrifying ideas that he rented office space in an old violin factory she owns, and together they are going their separate ways.

'We are continuing our individual careers, but we'll get together professionally on the apparel ideas so we can produce them commercially,' Riggs says.

The catch, of course will be to bring down costs and make the electronics equipment small enough to be comfortable and unobtrusive. Riggs and Bernstein say they can do both.

The two plan to get a business going somehow within the next few months. They have declined Lear's offer to buy the $6,000 shirts, since they want to keep prototypes as an aid to future planning. 'We invested our own money for the shirts and lent them to Lear for the show. We'll lend them again, whenever he needs them—and continue to do even more exciting electronic things for him if he'd like. But right now, we'd rather own the shirts than the $6,000.'

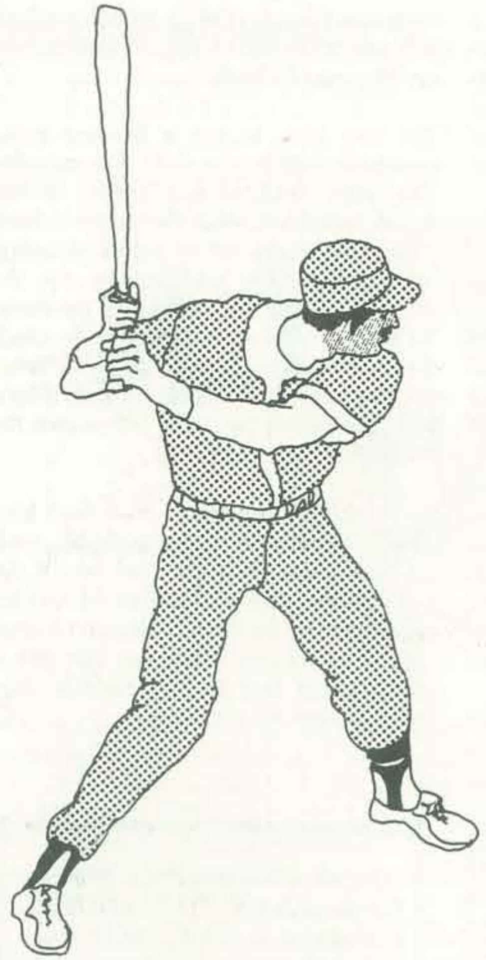
Their current project is a 24-foot kite. 'It is absolutely crushable, packable, wads up into a tiny ball. But you take it out and it goes "thoop"—and all 24 feet are right up there in the air,' Berstein beams. 'Yes, and we can make one just like it that's 1,000 feet long—a flexible, sky-borne sculpture,' Riggs adds. □

'Esoteric Computerabilia' is a new feature in PEOPLE'S COMPUTERS. Our aim is to keep you in touch with the most eye-catching, ear-bending, mind-boggling, fancy-tickling (heh-heh) far-out computable gossip that passes through our multi-sensory channels from time to time.

To this end, we are seeking your in*put! Save those newspaper clippings, documented rumors, photographs, catalogue items, unusual form letters—whatever computer/micro-electronic tidbits that you think might electrify our readers—note dates and sources, and send them to:*

Esoteric Computerabilia
c/o PEOPLE'S COMPUTERS
P.O. Box E
Menlo Park, CA 94025

We will, of course, be pleased to give you credit if we publish your tidbit.



BASEBALL

BY GARTH DOLLAHITE

Garth Dollahite, a senior at Franklin High School in Stockton, CA, has written a baseball game in HP2000 BASIC for his school computer. The computer is shared with two other high schools; each school has about 2400 students. Garth reports that BASE has become the most popular computer recreation on the system. Records kept of the number of executions of all the computer games indicate that BASE has been used twice as often as the second most popular game.

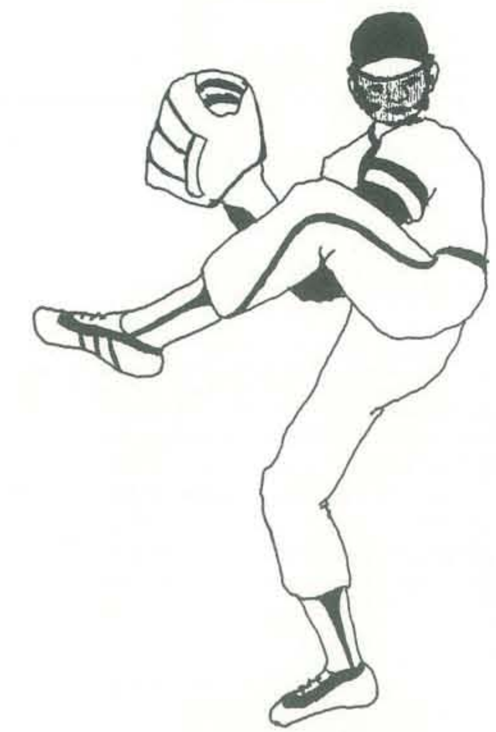
Dot shows which were swung at.

Any character followed by 'RETURN' will give score, outs, and a 9-second pause.

```
INTRODUCTION?N
WHAT IS YOUR NAME?FRANK
BASES CLEAN

 PITCH OF .111   BALL
 PITCH OF .016   BALL
 PITCH OF .814  .HIT...SINGLE
MAN ON  FIRST

 PITCH OF .608  .HIT...SINGLE
MEN ON  FIRST   SECOND

 PITCH OF .041   BALL
 PITCH OF .633  .STRIKE
 PITCH OF .465   BALL
 PITCH OF .589  .STRIKE
 PITCH OF .657  .STRIKE
OUT NUMBER 1    SCORE = 0

MEN ON  FIRST   SECOND

 PITCH OF .719  .STRIKE
 PITCH OF .471   BALL
 PITCH OF .353   STRIKE
 PITCH OF .793  .HIT...FOUL
 PITCH OF .932  .HIT...SINGLE
BASES LOADED

 PITCH OF .893  .STRIKE
 PITCH OF .445   BALL
 PITCH OF .564   BALL
 PITCH OF .546   STRIKE
 PITCH OF .270   BALL
 PITCH OF .350   STRIKE
OUT NUMBER 2    SCORE = 0

BASES LOADED

 PITCH OF .319   STRIKE
 PITCH OF .576  .STRIKE
 PITCH OF .980  .HIT...DOUBLE
MEN ON  SECOND  THIRD

 PITCH OF .277A
  SCORE = 2    OUTS = 2    READY?
 PITCH OF .277   BALL
 PITCH OF .981  .HIT...FOUL
 PITCH OF .696  .HIT...FOUL
 PITCH OF .692  .HIT...SINGLE
BASES LOADED

 PITCH OF .846  .HIT...FOUL
 PITCH OF .944  .HIT...SINGLE
BASES LOADED

 PITCH OF .332   STRIKE
 PITCH OF .234   BALL
 PITCH OF .418   STRIKE
 PITCH OF .428   BALL
 PITCH OF .476   STRIKE
OUT NUMBER 3   SCORE = 3


 TOP TEN SCORES OF 1157    EXECUTIONS

  24   GARTH
  23   MR. GAMES
  21   GARTH D.
  21   JOHN
  18   JODY
  17   RANDY THE KID
  17   GARTH D.
  17   THE KID
  17   MIKE D
  17   FRANK
  17   DAVID

 ANOTHER INNING?N
```



```
10  REM ***     BASEBALL BY      ***
20  REM ***   GARTH DOLLAHITE    ***
30  REM *** STOCKTON CALIFORNIA ***
40  REM ***      JUNE 1977       ***
50  DIM B$[18],N$[15]
60  PRINT "INTRODUCTION";
70  INPUT X$[1,1]
80  IF X$="N" THEN 190
90  PRINT "THIS IS A SOLITARY BASEBALL GAME"'10
100  PRINT "I WILL PITCH THE BALL TO YOU.  I WILL RATE THE PITCH BETWEEN"
110  PRINT "ZERO AND ONE, ONE BEING PERFECT, ZERO THE PITS, SO .500 IS"
120  PRINT "AN AVERAGE PITCH.  TO SWING AT THE BALL, HIT 'RETURN' AFTER"
130  PRINT "I GIVE THE NUMBER.  IF RETURN IS NOT HIT WITHIN ONE SECOND,"
140  PRINT "YOU ARE LETTING THE BALL GO BY (WILL BE A STRIKE OR BALL)"'10
150  PRINT "THIS GAME RUNS CONTINUOUSLY.  THE ONLY TIME I WILL PAUSE"
160  PRINT "WILL BE IF YOU TYPE A CHARACTER (FOLLOWED BY 'RETURN')"
170  PRINT "ON YOUR FIRST SWING AFTER I TELL YOU WHO'S ON BASES."'10
180  PRINT "YOU COMPETE AGAINST PREVIOUS PLAYERS FOR HIGH SCORES"'10
190  PRINT "WHAT IS YOUR NAME";
200  INPUT N$
210  B$="FIRST SECONDTHIRD"
220  MAT B=ZER
230  H=0
240  FOR O=1 TO 3
250  S=B=0
260  GOTO B[1]+B[2]+B[3] OF 380,400,290
270  PRINT "BASES CLEAN";
280  GOTO 450
290  PRINT "BASES LOADED";
300  GOTO 450
310  IF V THEN 590
320  IF Q=-256 THEN 590
330  PRINT '10"  SCORE ="H"OUTS ="O-1"READY?";
340  V=1
350  ENTER 9,W,Z
360  PRINT
370  GOTO 550
380  PRINT "MAN ON  ";
390  GOTO 410
400  PRINT "MEN ON  ";
410  FOR X=1 TO 3
420  IF  NOT B[X] THEN 440
430  PRINT B$[X*6-5,X*6]"  ";
440  NEXT X
450  PRINT '10
460  V=0
470  P=RND(1)
480  IF P>.0005 THEN 520
490  PRINT "YOU'VE JUST BEEN BEANED WITH THE BALL"
500  PRINT "CRAWL TO FIRST"
510  GOTO 740
520  IF P<.9995 THEN 550
530  PRINT "PAY MORE ATTENTION, IT JUST WIZZED BY YOU"
540  GOTO 650
550  PRINT  USING 560;P
560  IMAGE #," PITCH OF ",.3D
570  ENTER 1,Q,Z$[1,1]
580  IF Z$#" " THEN 310
590  P=P+RND(1)
600  V=1
610  PRINT '13TAB(16);
620  IF Q#-256 THEN 810
630  PRINT " ";
640  IF P<1 THEN 700
650  PRINT "STRIKE"
660  S=S+1
670  IF S<3 THEN 470
680  PRINT "OUT NUMBER"O"SCORE ="H'10
690  GOTO 1110
700  PRINT "BALL"
710  B=B+1
720  IF B<4 THEN 470
730  PRINT "WALK"
740  FOR X=1 TO 3
750  IF B[X] THEN 780
760  B[X]=1
770  GOTO 250
780  NEXT X
790  H=H+1
800  GOTO 250
810  PRINT ".";
820  IF P<1 THEN 650
830  PRINT "HIT...";
840  IF P>1.5 THEN 990
850  GOTO INT(RND(1)*8) OF 890,890,890,910,1000
860  PRINT "FOUL"
870  IF S=2 THEN 470
```



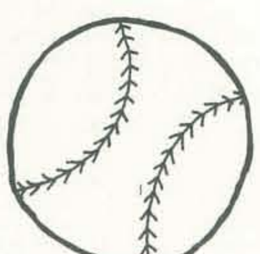
You are the batter for half an inning of baseball. You will be given the choice of when to swing. You will be at the plate throughout the game. The object is to get as many runs as possible in the inning.

Pitches are two random numbers (each from zero to one, non-inclusive). The computer will print the first number as the pitch. You will then have one second to decide whether to swing or not. If the sum of the two numbers in greater than one, the pitch is good. If the sum is greater than 1.5, a hit will have a higher potential for extra bases. The game continues until three outs are made.

Before running the program, create two files, BALL and BALL1. These will be used to keep a record of the best scores of all previous players. If you do not want the files' option, see lines 1120 and 1130 of the program.

baseball

```
880  GOTO 660
890  PRINT "SINGLE"
900  GOTO 740
910  PRINT "DOUBLE"
920  B[3]=B[1]+B[2]+B[3]
930  IF B[3]<2 THEN 960
940  H=H+B[3]-1
950  B[3]=1
960  B[1]=0
970  B[2]=1
980  GOTO 250
990  GOTO INT(RND(1)*15) OF 890,890,890,890,910,910,910,1020,1020,1070
1000  PRINT "OUT"
1010  GOTO 680
1020  H=H+B[1]+B[2]+B[3]
1030  MAT B=ZER
1040  B[3]=1
1050  PRINT "TRIPLE"
1060  GOTO 250
1070  H=H+B[1]+B[2]+B[3]+1
1080  MAT B=ZER
1090  PRINT "HOME RUN"'7
1100  GOTO 250
1110  NEXT O
1120  REM *** LINES 1140 THROUGH 1530 MAY BE DELETED ***
1130  REM *** IF FILE ACCESS IS UNAVAILABLE         ***
1140  K=BRK(0)
1150  FILES BALL,BALL1
1160  LOCK #1,Q
1170  IF Q THEN 1160
1180  IF  END #1 THEN 1200
1190  GOTO 1210
1200  PRINT #1,1;0
1210  READ #1,1;P
1220  READ #2,1
1230  Z=U=N=T=0
1240  IF  END #1 THEN 1330
1250  READ #1;B$,B
1260  Z=Z+1
1270  IF B<H AND  NOT U THEN 1300
1280  PRINT #2;B$,B
1290  GOTO 1250
1300  PRINT #2;N$,H,B$,B
1310  U=1
1320  GOTO 1250
1330  IF B#H THEN 1350
1340  PRINT #2;N$,H
1350  READ #1,1
1360  UPDATE #1;P+1
1370  IF  NOT U AND B#H AND Z>9 THEN 1460
1380  READ #2,1
1390  IF  END #2 THEN 1460
1400  READ #2;B$,A
1410  IF N>9 AND A<T THEN 1460
1420  PRINT #1;B$,A
1430  N=N+1
1440  T=A
1450  GOTO 1390
1460  UNLOCK #1
1470  K=BRK(1)
1480  READ #1,1;P
1490  PRINT '10'10"TOP TEN SCORES OF"P"EXECUTIONS"'10
1500  IF  END #1 THEN 1540
1510  READ #1;B$,A
1520  PRINT A;B$
1530  GOTO 1510
1540  PRINT '10"ANOTHER INNING";
1550  INPUT X$[1,1]
1560  IF X$="Y" THEN 210
1570  END
```



LETTERS, cont'd

The following two letters are requesting information that may be of interest to many of our readers. We will be happy to publish information on these topics either as letters or as full articles.

I am focusing on interactive technologies and their implications for education and community participation in public policies. My perceptual frame is Constitutional—the First Amendment right to communicate in contemporary technological terms, and Health (mental and physical)—the fulfillment of self-esteem/self-efficacy needs through responsible citizen participation in governance, and Evolutionary Adaptation with Civility—meaning, how do well-meaning people make these political choices, given the state-of-the-art of information and communication data, science and technology?

Specifically, I am designing a civic leaders seminar for updating awareness and information on the implications of 'The Information Age' for the internal and external communications of non-profit civic organizations. But, I need help. First, I need to know of models of uses people are making of computers and computer networks for public policy information design, information gathering, processing, storage and retrieval.

Next, is anyone yet up to complex processing to understand human and social impacts, tradeoffs among interdependent policies and long term consequences? If so, I'd like to be in touch with them. Last, I would like to know of people in communities who are developing their own education software, conventional or computer-assisted.

Minna Post Peyser
New York, NY

:::

I am in the process of researching homemakers, working women and 'workwives' and the impact of technology on both them and their families. One area of particular interest to me is how computers (Home use) and computer-based equipment will be marketed to the homemaker. Do you have or know of any sources on the subject?

Kathleen Hardiman Arnold
Commission on The Status of Women
San Francisco, CA □

TRS-80

BY CLYDE FARRELL

CONVERTING TO LEVEL II BASIC

Clyde Farrell is one of our strongest and most vocal TRS-80 supporters. Last issue, we published some of his comments as part of a growing dialogue on the TRS-80 (see the Letters section of this issue) as well as his TRS-80 WUMPUS program.

Radio Shack has just recently released their long-awaited Level II BASIC, and Clyde's comments on converting Level I programs to Level II are thus timely and most welcome. Also, his DRAW program will give you a fun way to try out Level II BASIC. Clyde has written a number of games (including WUMPUS, STAR TREK and some epic games) for the TRS-80 in both Level I and Level II BASIC. These are available through:

Farrell Enterprises
P.O. Box 4292
Walnut Creek, CA 94596

–BK

Level II BASIC at last!! The long awaited successor to Level I has finally arrived, and the improvement in capability and performance is truly awesome. With the addition of string handling ability, multi-dimensional arrays, editing and program execution tracing features, new programs are easier than ever to develop and much more interesting to use. Even old Level I programs are improved, since they now load and run much faster than before.

Level I programs won't load and run, however, before they undergo a conversion routine that may be a little more involved than what you have been led to believe.

The Level II conversion package I received from Radio Shack contained two cassette tapes and a few pages of documentation. One cassette was for converting data tapes. The documentation I received explained that the conversion process consisted of three steps, namely: 1) load the conversion program using the 'SYSTEM' command, 2) load the Level I tape to be converted and 3) start execution of the conversion program. All of this worked very well, but when I typed 'RUN' I began to discover some of the differences between the two BASICs.

The first thing to happen in the particular program I chose to convert (STARTREK) was not totally unexpected. The video display flashed 'BS ERROR IN LINE 220'. After consulting the manual, I determined that I had gone *B*eyond *S*ubscript in an array. This was due to the fact that Level I would dynamically assign space to its single A(x) array until memory had been exceeded, whereas Level II requires the more conventional DIMensioning of arrays. This problem was discussed in the conversion documentation, and with the addition of a DIM statement to the program, the problem was corrected.

I typed 'RUN' again and found, very much to my joy, that the amount of time I had to wait while the program initialized was about half of what I was used to. I was even more surprised, however, when instead of seeing the symmetrical group of K's, E's, B's and dots I expected, I saw an unintelligible mess spread out all over the screen. BREAKing the run, I found a peculiarity that had not been mentioned in the conversion documentation. In Level I BASIC, the statement

10 PRINT TAB(25),"K"

with a comma following the 'TAB(x)' was the only way a 'PRINT TAB(x)' statement would be accepted by the interpreter. Level II differs here as it will accept this line with a comma, a semi-colon, or with nothing at all. The effect of the comma, though, is the same as if the 'TAB(x)' portion was left out altogether; and it caused everything to be printed in four widely spaced columns down the face of the screen. After editing the commas to semi-colons, the familiar eight-by-eight grid reappeared.

Everything worked quite well for awhile until the first time I encountered a Klingon. Again, the program crashed and I was now confronted with another cryptic message: 'TM ERROR IN 540'. I retreated to my manual to find I had a 'Type Mismatch' error. After some deliberation and hard staring at Line 540, I finally realized another small quirk between the two BASICs. Level I allowed statements like

540 A$=KLINGON

without the need for using quotation marks around 'KLINGON', but Level II BASIC insists upon it. Since the conversion program will not insert these quotes for you, each Level I program you convert that uses the string variables will have to have the quotes added.

With this final correction, the program proceeded to run extremely well and with a new-found speed that truly added to my overall enjoyment. No other conversion problems turned up after the discovery of these few. A couple of other unexpected conditions did surface, though, while typing new programs into Level II; and both involve the use of the 'shift' key.

In Level I BASIC, the programmer had the use of the PRINT AT statement. In Level II, the effect of the command is the same but the format is changed to PRINT@. This format change can be noticed in programs converted from Level I to Level II where the conversion programs made the change for you. The condition to be careful of when typing is that PRINT@ is not considered to be the same as PRINT shift@, *even though the '@' appears on the video monitor the same way*. The '@' key does not have the upper and lower case characters on it as do the other special character keys, and the user may be tempted to assume that the shift key would not affect operation. It was very difficult to determine the bug in the program the first time this happened since the statement *looked* completely right. The shifted '@' causes the program to crash and give you a syntax error.

The other related condition is also very elusive. It occurs when assigning a value to a string variable. Consider, for example, the lines:

10 INPUT A$
20 IF A$="Y" GOTO 1000

If, when line 20 is typed in, the user does not release the shift key when typing the 'Y' between the quotation marks, the program will look for a shift 'Y' during execution to satisfy the condition. In other words, if during execution of the program, the user simply types a 'Y', the program *will not* branch to line 1000. Then when the programmer lists the line to determine what his error was, the line appears to be just fine!! Believe me, this can be very mystifying when you don't know what's happening, and very useful when you do!!

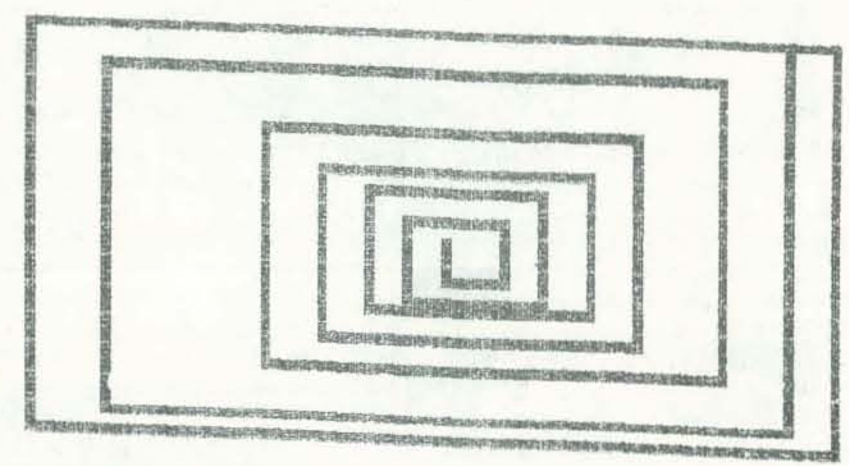
Level II is a great improvement to the TRS-80, and with these subtle differences from Level I in mind, your conversions should flow very smoothly.

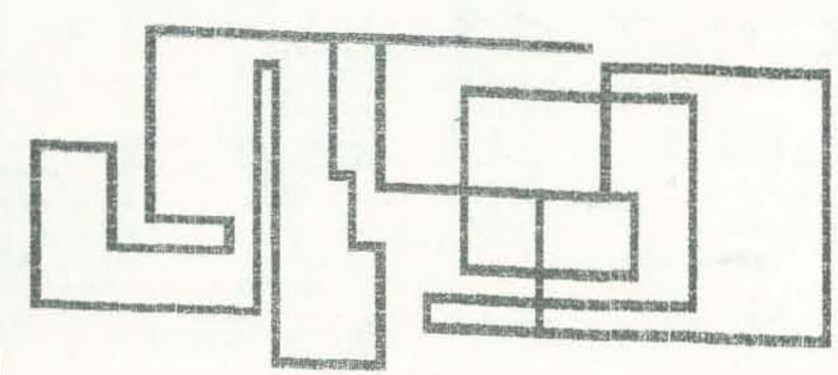
DRAW

This short program gives TRS-80 owners with Level II BASIC the ability to use their computer's graphics under keyboard control. When typing this program into the machine, it is very important to remember a little 'trick'. In lines 35, 45, 55 and 65, the letter contained within the quotation marks must be typed using the shift key. All other letters within quotes should be typed without the use of the shift key. Although the shifted and unshifted letters will appear the same on the screen, the byte stored in memory is different. (See article.)

To use this program, type 'RUN' and the screen will clear leaving only a flashing 'cursor'. To draw a line to the right, simply press 'R' on the keyboard. It is not necessary to press the enter key. To stop drawing, press the space bar. To draw up, down or left simply press the first letter of the direction desired. Erasing a line or positioning the 'cursor' is accomplished by pressing both the shift key and the appropriate letter–U, D, L or R. Depressing the 'C' will clear the screen again. Enjoy!

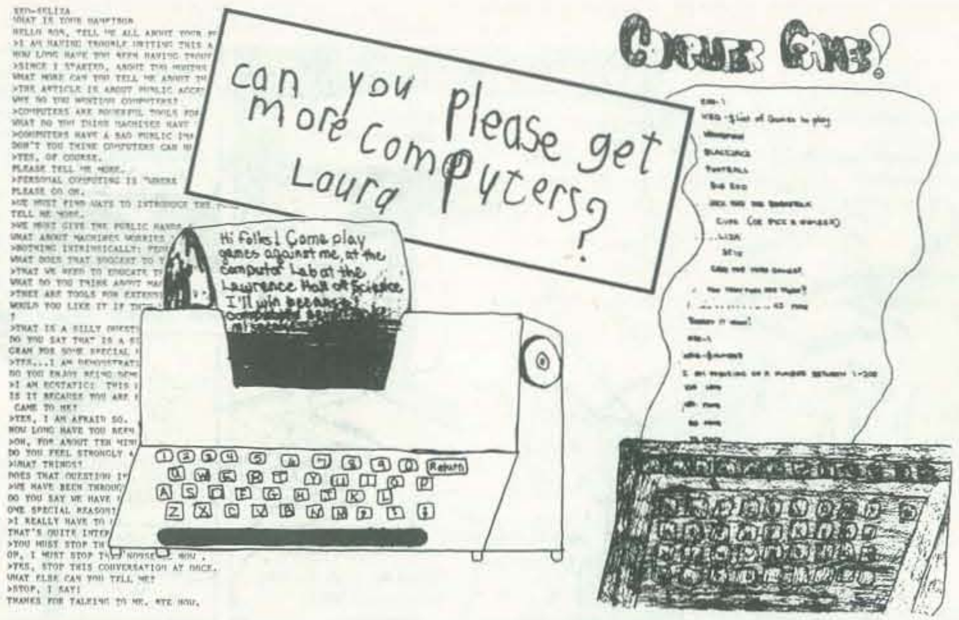
```
5 CLS
10 X=1:Y=1
20 B$=INKEY$
21 IFB$<>""THENA$=B$
30 IFA$="L"THENX=X-1:GOSUB1000
35 IFA$="L"THENRESET(X,Y):X=X-1
37 IFX<0THENX=0
40 IFA$="R"THENX=X+1:GOSUB1000
45 IFA$="R"THENRESET(X,Y):X=X+1
47 IFX>127THENX=127
50 IFA$="D"THENY=Y+1:GOSUB1000
55 IFA$="D"THENRESET(X,Y):Y=Y+1
57 IFY>47THENY=47
58 IFA$="C"THENCLS
60 IFA$="U"THENY=Y-1:GOSUB1000
65 IFA$="U"THENRESET(X,Y):Y=Y-1
67 IFY<0THENY=0
68 P=POINT(X,Y)
69 SET(X,Y):FORS=1TO30:NEXT:RES
ET(X,Y):IFP=-1THENSET(X,Y)
70 GOTO20
1000 IFX<0THENX=0:RETURN
1001 IFY<0THENY=0:RETURN
1002 IFX>127THENX=127
1003 IFY>47THENY=47
1010 SET(X,Y):RETURN
```






FORTRAN MAN
BY LEE SCHNEIDER & TODD VOROS
Volume III, Episode 3
As you may recall from our last episode, Fortran Man was in execution of a plan to relocate himself into Microprocessor Land—otherwise known as the Land of the Little People—where his help is desperately needed by the Underground Resistance Movement, who are at this microsecond engaged in fierce combat with the evil Glitchmaster!
Since the Glitchmaster had stolen the Lockout Monster from its rightful owners in Clan McIntel, all standard channels of communication into and out of the land are inhibited; and therefore Our Hero was forced to compile a plan which involved a more unusual form of travel—in PROM!
With the aid of Firmware Interface Control, Fortran Man was transferred onto PROM, whereupon his old friend and fellow crime-fighter, Billy Basic, was to smuggle him past the gates and into Microporcessor land, there to be reloaded from PROM and back into execution!
But then . . . the unexpected! Once at the gates, a suspicious Data Security Guard attempted to expose the PROMs to ultraviolet light! Billy grabbed Our Hero and ran for it! But he was not fast enough . . .
The Guards caught him . . . and in the struggle, the chip carrier containing the PROMs fell into the data channel and was swept away by the currents . . . and Billy Basic, accused of data smuggling, was stripped of his line numbers and placed in HOLD state—indefinitely!
Whew! Moving these heavy-current supplies is hard work!
Yeah, how come we gotta do all the work just because you're 5% and we're 10%?
Because I got more stripes, that's why! Now get moving before you exceed my tolerance!
Meanwhile, far away in the central highlevels of Microprocessor Land, a small detachment of the Underground Resistance Movement is transporting supplies to their secret rebel base . . .
They journey onward . . . then, suddenly, they come upon a hidden valley nestled among the higher level logics, and within that valley are mounted the components of the rebel camp . . .
It is not the largest of the rebel camps, but it is an important one, for this branch of the resistance is commanded by none other than Linea, who throughout the war has proven to be one of the greatest thorns in the Glitchmaster's side . . . and even now she is plotting with her compatriots to hamper his evil plots all the more . . .
As you can trace on this memory map, gentlemen, the Glitchmaster plans to move against us by propagating across the channel at this point! Therefore we shall send our resistance to unbalance the bridge, and shunt that current movement to ground!
Yes, I see! If only it were not for the Lockout Monster keeping us out of the core planes, we could attack and destroy him utterly!
In the midst of that meeting comes a sudden Interrupt . . .
Commander! The supplies have arrived from the voltage vendors . . . and something else!
One of our patrols was watching along the banks of the data channel, and they spotted this floating by!
Hmmm . . . looks like a chip carrier! I wonder what's in it?
Hmmm . . . PROMs! Unmarked, too . . . could be secret dispatches from the Glitchmaster!
They seem intact; . . . if only we had some way of dumping them!
puff puff
But we do! In our last assault on the system bus we captured a number of signals—including a loader! Its only a relocatable field model, but I think it should work!
On Linea's command, the carrier is placed upon the data table, the seal is broken and the case opened . . .
In her usual manner, Linea wastes no cycles in having the PROMs transported to the field loading unit, where preparations are made to transfer the contained data back into executable form . . .
I shall need some supplies to operate the device; are there any available?
Yes, but of what? That doesn't look like any piece of standard microprocessor code to me!
I believe some just arrived—they will be sent for!
Aha! Just as I predicted—a successful reload!
ZAP
We must question him!
Supplies are brought and connected as ordered . . . and after a short delay for address checking, the PROMs are inserted and the LOAD function initiated . . .


And question him they do . . .
All right, who are you, where did you come from, and what are you doing here?
Huh? Er . . . what? Why, I'm . . . er . . . I'm . . .
Come on . . . out with it! Surely you know who you are . . . tell us!
I . . . er . . . I can't seem to remember!
Where am I? Or for that matter, who am I?
drawing by amiya
I think it's obvious, commander! I thought there was something peculiar about those PROMs—and now I know what it was! They must have been exposed to UV light. Part of the poor fellow's memory has been erased!
The question is, though, do we trust him? Or branch him back into the channel?
Hmmm . . . seems harmless enough. Still . . .
Who? What?
Commander!
Then suddenly the conversation is interrupted by an important breakpoint messenger . . .
Commander! Our position has been decoded, and we are being attacked!
Two columns of Random Numbers have been detected on the data fields above us and even now they are forming a converging series towards our position!
Swiftly, configuration orders are issued . . .
Quickly! Have the troops form series lines to increase our resistance . . . or we will be overflowed!
Invert polarity on our transformer squad and bring up the secondaries!
And . . . hey! Where did that stranger go?
Look!
The entire camp turns to look in amazement, for as the enemy colums shift downwards into the valley, a single lone figure is branching upward with incredible speed to meet them head-on . . .
Good grief! What does he think he's up to??
ZONK!
POW
CRUNCH!
SOCK!
And within microseconds, the battle is on! Without warning the attacking columns are assaulted from all sides at once, and the oscillations propagate from one end of the valley to the other . . .
And before any of the resistance fighters can even respond to the CALL to arms, the random numbers have been put into order, the battle is over, and the mysterious stranger does a rapid RETURN! The whole camp turns out to greet him . . .
Amazing! Fantastic!
Defeated an entire regiment of the Glitchmaster's best troops—single-handedly!
How . . . how did you do it????
Wha . . . you mean . . . did I do that?
I . . . I don't know! I just felt compelled to compile!
But the highest praise comes from Linea herself . . .
No matter how you did it, friend—for now it is certain—you are one of us!
And now, with your abilities, perhaps we will be able to accomplish what, for so long, we have tried and failed to accomplish . . . Gentlemen: Gather your troops, and prepare yourselves! On the rising edge of the next cycle, we attack the stronghold of the Glitchmaster!
Yes, Commander!
And so, dear readers, the war is just beginning. But, will Fortran Man ever recover his lost memory area? Or is he destined to remain an unidentified file? Will the Underground Resistance movement triumph over the Glitchmaster, or will the Lockout Monster destroy them? Will Billy Basic ever be free to RUN again? Tune in again next issue—same timeslice, same channel number. . . .



Computers and Science Museums: A Public Access Model

BY BOB KAHN

To fully achieve the goal of public computer literacy, we must go beyond books, magazines and films about computers; people must have hands-on access to computers in a personal context. How can such public access be made available? Where, besides computer fairs and stores, can people go to learn about and play with computers? One answer: a number of science museums or science-technology centers all across the country. This is the first of a three-part serialized article exploring the ways in which these centers are making computers accessible and understandable to the general public. An earlier version of this article was published in IEEE Computer Magazine, *April 1977. We are grateful to* IEEE *for granting us permission to reprint sections of the original article and, in particular, the figures and tables. –BK*

MUSEUMS AND COMPUTERS

Traditionally, one thinks of the museum as a repository for stuffed animals, art and artifacts from the present and past. The science museum is often a maze of exhibits (frequently automated by push-buttons and floor mat switches) illuminating various aspects of science, history or technology. Likewise, a popular mass media image of the computer depicts a row of magnetic tape drives lining the walls of an air-conditioned, false-floored, glassed-in room where specially trained computer operators and programmers huddle around a central console of blinking lights. Ultimately, the (traditional) computer will undoubtedly come to rest as an artifact in the (traditional) museum of science and technology, as for example, ENIAC, the first general-purpose electronic computer now rests in the Smithsonian Institution.

From an educational standpoint, both the museum and early generations of computers, whose image is depicted above, have shared a common characteristic: they have tended to be static and passive. They have offered the public little opportunity for participation or access to technology. A museum visit, while being a very worthwhile and culturally respectable way to spend a Saturday afternoon, often involves a lot of walking, looking and listening, but little touching or participation (I purposely exclude button-pushing to turn on an exhibit as a participatory activity).

For those who do not already use computers in business, school or the university, public access has generally been vicarious or limited to computer output only. We watch the airline ticket agent make a reservation at the terminal, we receive computer-generated form letters and bills in the mail and we peer through the glass portal at the computerized San Francisco BART (*B*ay *A*rea *R*apid *T*ransit) Control Room.

However, computer technology has advanced considerably beyond what the public has experienced and science-technology centers are re-defining their traditional role as a community showcase and repository. This article discusses ways in which these science-technology centers are making modern computers accessible and understandable to the general public.

Based on material originally appearing in 'Public Access to Personal Computing: A New Role for Science Museums' from *COMPUTER Magazine*, April, 1977. Reprinted with permission of IEEE.

The Power of Personal Computing and the Need for Public Computer Literacy. Slowly, people are becoming aware of the potential power computers have in areas beyond business and data analysis. With good software, the computer can become a personalized tool for extending the mind: a personal laboratory for simulated experimentation, an artistic medium, an information retrieval system, a gaming opponent and personal entertainment center, a teacher, a sophisticated editing facility and of course, a programmable calculator–all in the same box. This is the essence and the 'power' of personal computing. When people (particularly children) have been exposed to personal computing, they search out and demand access to it.

This demand is evident in the proliferation of homebrew computer clubs, microcomputer kits, 'Byte shops,' and computer hobbyist newsletters. But awareness of personal computing must spread far beyond the hobbyist movement.

While one no longer needs a crystal ball to foresee a computer in every home, it will still be a number of years before everyone's television becomes a terminal. In those intervening years, it is imperative that adults and children, not just professionals and hobbyists, achieve a reasonable level of computer awareness or 'computer literacy.' Computer literacy has been well defined in a recent report by Amara[1] for the Institute for the Future as 'the understanding of basic computer functions in terms of what computers can and cannot do, with particular attention to their potential, as well as their limits, in meeting human needs.' While many recent books and films have attempted to speak to this need, computer literacy can be quickly and effectively achieved by giving the general public direct, hands-on access to computers in a non-threatening, relaxed learning environment. Where, then, do such environments exist for the general public to gain access to personal computing power?

Science Centers for the Public. One type of institution that is ideally suited to provide public access to computing is the science-technology center–not the traditional science museum described earlier, but an emerging new type of activity-oriented community learning resource devoted to furthering the public's appreciation for the methods, tools and principles of science and scientific research.

There are nearly 300 science and natural history museums in the U S, Canada, and Mexico. Of these 54 are currently listed as either full or associate members of the Association of Science-Technology Centers (ASTC), founded in 1973. These institutions either have, or are developing, extensive educational programs, specifically, this includes having 'a substantial number of exhibits, demonstrations and programs designed to further public understanding and appreciation of science; being inter-disciplinary in nature, with emphasis on the physical and life science; being non-profit and open to the public; and using visitor participation techniques.'[7] In 1975, these 54 ASTC-affiliated centers alone drew more than 32.5 million visitors, including some 9 million children visiting as part of school groups. Unlike the traditional museum, the hallmark of science-technology centers is 'discovery learning' or 'learning science by doing science.' Instead of rows of glass cases protected by signs and security guards, modern science centers encourage visitors to touch objects, manipulate controls, participate in workshops and classes in an informal, non-school museum environment. Clearly, interactive computers and computer terminals would be right at home in such places. Indeed, most science-technology centers recently built or still in the planning stages have made provisions for computer-related exhibits and activities. However, few of the older centers (let alone science museums) have made computing accessible to the public, although many have expressed interest in doing so. Not surprisingly, bringing about change in established institutions, even institutions which are concerned with the latest technology, can be a slow, difficult and expensive undertaking.


Interactive computer games provide an enjoyable challenge to young minds.

PUBLIC ACCESS TO COMPUTING THROUGH SCIENCE-TECHNOLOGY CENTERS

Early in 1976, ASTC published a survey of educational programs at its member institutions[3]. Of the 54 centers surveyed, 27 returned the questionnaire, and of that 27, only 10 institutions had one or more computer terminals at that time.* This situation is changing rapidly with increasing availability of inexpensive, self-contained micro-computer systems. To describe the involvement of these institutions in making the public aware of the potential of computers, the ladder-like model is shown in Figure 1. The model contains five levels and is stratified so that each higher level usually includes all those levels below it as well and most–if not all–science-technology centers and museums may be located on an appropriate level. The various levels of the model are described moving from minimal public involvement at the bottom to potentially attainable involvement at the top, citing examples of institutions whose programs are particularly exemplary within various levels.

*Note that because only half of the ASTC members responded to the survey and because some institutions having a substantial computer component such as the Boston Children's Museum are not members of ASTC, these summary statistics and the profiles which follow must be viewed as exemplary rather than as exhaustive. While I have no doubt that the activities and programs described are quite representative of the status quo, I may well have omitted some institutions having similar, or even more extensive, computer-related programs.

Levels	*Characteristics*	*Examples*
LEVEL IV: Ideal Public Access Computing Center	Basic Research on the Person-Computer Interaction Development of New Personal and Public Uses of Computers Community Program Library and Data Bank Publicly Accessible Showcase for Newest Technology Development and Dissemination of Educational Computer Programs, Resource for Personal & Home Computing Needs	
LEVEL III: Active Community Computer Education	Community Computer Education Resource and Time-Sharing Service Range of Classes for Children & Adults Many Terminals Located in Exhibit Area & Computer Terminal Rooms (Pay by the Hour) Use Library Programs or Write Own Programs	Lawrence Hall of Science
LEVEL II: General Public Access	Educational Computing Resource Center Applications Programs Written & Distributed On-Site Computer & Computer Specialist 1 or More Interactive Terminals in Exhibit Area with Repertory of Programs, Games, Information, etc.	The Boston Children's Museum Oregon Museum of Science & Ind. Boston Museum of Science Maryland Academy of Sciences Ontario Science Centre Pacific Science Center
LEVEL I: Special Public Access	A Programming Class for Local Kids School Field Trips and/or Special Classes, Workshops & Demonstrations for School Children, ages 6-18 One or More Computerized Exhibits Special Kids in the Basement	Franklin Institute Science Museum & Planetarium California Museum of Science & Ind. Chicago Museum of Science & Ind. Fernbank Science Center
GROUND LEVEL: Potential Public Access	Research, Data Processing, Cataloging and Record-keeping, Visitor Scheduling, and Controlling Equipment	A Majority of Science-Technology Centers and Museums

Figure 1. A model of public access to computers through science-technology centers and museums.

Starting at the bottom of the model in Figure 1, the ground level or 'lowest common denominator' for computer use in science centers and museums is research, data processing, cataloging and record-keeping, visitor scheduling and controlling equipment such as the museum's planetarium. Like other businesses, museums have information processing needs. Thus, when a museum or science center acquires a computer or computer services, this is often one of the first, and sometimes the only, application to be realized. (The April 1973 issue of *Museum News* was dedicated to the various uses of computers in museums. Seven of the eight articles in that issue dealt with these information processing aspects.)

In a sense, this level does not belong in the model; our concern here is with public access to computing and not just with the ways in which these institutions make use of computing power. However, an institution that is using computers for its own business needs can potentially begin to explore ways to make computing available to the public. Therefore, this might better be called the seed level of the model; once an institution acquires some access to computing power, it has the option to decide to what extent this power will in turn be accessible to the public.

A museum that builds a glass case around the computer for the public to watch the machine at work, and not interact with it, is reinforcing the mass-media image of computers. This is not really a commitment to public computer access. In such cases, it might be better to keep the machine hidden and say nothing of its existence. Instead, an institution must begin to exhibit its computing power in such a way that the general public may not only watch the machine work, but also work (play) with the machine. Indeed, an institution that is truly committed to educating the public about computers and their use obtains a computer specifically and primarily for this purpose. Such an institution treats administrative data processing and record-

ELIZA TALKS TO BOB ABOUT PUBLIC ACCESS.

```
XEO-$ELIZA
WHAT IS YOUR NAME?BOB
HELLO BOB, TELL ME ALL ABOUT YOUR PROBLEMS.
>I AM HAVING TROUBLE WRITING THIS ARTICLE.
HOW LONG HAVE YOU BEEN HAVING TROUBLE WRITING THIS ARTICLE ?
>SINCE I STARTED, ABOUT TWO MONTHS AGO.
WHAT MORE CAN YOU TELL ME ABOUT THAT?
>THE ARTICLE IS ABOUT PUBLIC ACCESS TO COMPUTERS.
WHY DO YOU MENTION COMPUTERS?
>COMPUTERS ARE POWERFUL TOOLS FOR EXTENDING OUR MINDS.
WHAT DO YOU THINK MACHINES HAVE TO DO WITH YOUR PROBLEM?
>COMPUTERS HAVE A BAD PUBLIC IMAGE.
DON'T YOU THINK COMPUTERS CAN HELP PEOPLE?
>YES, OF COURSE.
PLEASE TELL ME MORE.
>PERSONAL COMPUTING IS "WHERE IT'S AT"
PLEASE GO ON.
>WE MUST FIND WAYS TO INTRODUCE THE PUBLIC TO PERSONAL COMPUTING.
TELL ME MORE.
>WE MUST GIVE THE PUBLIC HANDS-ON ACCESS TO COMPUTERS.
WHAT ABOUT MACHINES WORRIES YOU?
>NOTHING INTRINSICALLY: PEOPLE'S MISCONCEPTIONS WORRY ME.
WHAT DOES THAT SUGGEST TO YOU?
>THAT WE NEED TO EDUCATE THE PUBLIC ABOUT THE POWER OF COMPUTERS.
WHAT DO YOU THINK ABOUT MACHINES?
>THEY ARE TOOLS FOR EXTENDING MAN'S HORIZONS.
WOULD YOU LIKE IT IF THEY WERE NOT TOOLS FOR EXTENDING MAN'S HORIZONS
?
>THAT IS A SILLY QUESTION: I THINK THERE IS A BUG IN YOUR PROGRAM.
DO YOU SAY THAT IS A SILLY QUESTION YOU THINK THERE IS A BUG IN MY PRO
GRAM FOR SOME SPECIAL REASO
>YES...I AM DEMONSTRATING HOW TO MAKE YOU TALK NONSENSE.
DO YOU ENJOY BEING DEMONSTRATING HOW TO MAKE I TALK NONSENSE ?
>I AM ECSTATIC!  THIS CONVERSATION IS PRICELESS.
IS IT BECAUSE YOU ARE ECSTATIC THIS CONVERSATION IS PRICELESS THAT YOU
 CAME TO ME?
>YES, I AM AFRAID SO.
HOW LONG HAVE YOU BEEN AFRAID SO ?
>OH, FOR ABOUT TEN MINUTES.
DO YOU FEEL STRONGLY ABOUT DISCUSSING SUCH THINGS?
>WHAT THINGS?
DOES THAT QUESTION INTEREST YOU?
>WE HAVE BEEN THROUGH THIS LOOP BEFORE; I MUST GO NOW.
DO YOU SAY WE HAVE BEEN THROUGH THIS LOOP BEFORE YOU MUST GO NOW FOR S
OME SPECIAL REASON?
>I REALLY HAVE TO GO NOW.
THAT'S QUITE INTERESTING.
>YOU MUST STOP THIS NONSENSE NOW.
OH, I MUST STOP THIS NONSENSE NOW .
>YES, STOP THIS CONVERSATION AT ONCE.
WHAT ELSE CAN YOU TELL ME?
>STOP, I SAY!
THANKS FOR TALKING TO ME. BYE NOW.
```

keeping as a separate aspect, or even a side benefit, rather than as the main justification for having a computer. This latter philosophy becomes increasingly more dominant as we move to each higher level in the model.

LEVEL I: SPECIAL PUBLIC ACCESS

Special Kids in the Basement. Once a public institution has acquired a computer, it is difficult to keep the machine a secret. Neighborhood children (young and old, including some computer whiz kids and hobbyists) always seem to start coming around to check out the equipment and find out how, why and when the machine is being used—and by whom. Some of these interested people turn up quite frequently, and sooner or later, they seem to acquire privileges, perhaps in exchange for programming services rendered, whether solicited or not. (Some of them are occasionally given long-term projects in exchange for peace of mind on the part of whoever is in charge of the machine.)

I call this phenomenon 'Special Kids in the Basement.' Undoubtedly, every public computer installation in the country has its own group of these special kids, whether it be just individuals or an organized group or club. They represent a very limited form of public access, but not an uncommon one. In the museum environment, these kids are generally allowed to work during scheduled classes, after hours, or on a time-available basis. They always seem to attain an amazing proficiency in using computer jargon, and their deftness in working with the computer can be somewhat overwhelming to the uninitiated . . . and even to the museum staff. The programs that they write usually become a part of the museum's program library, and many of these kids eventually get hired by or through the institution as programmers.

'We got the WUMPUS!!' Only the hunt is simulated in this computer game; the excitement is real.

Some institutions, such as the University of California's Lawrence Hall of Science (LHS) in Berkeley and the Oregon Museum of Science and Industry (OMSI) in Portland, have developed special programs for these 'special kids.' For several years, Pete Rowe, the senior applications programmer at the Lawrence Hall has been offering a 'free' (i.e., no tuition charge) class to local students who have become proficient in programming and who have a useful and interesting programming project they wish to complete. To be admitted to the class, students must submit a written proposal outlining the nature and goals of their project. Students whose proposals have been accepted gain specially arranged computer access on Friday afternoons and space-available access at other times during the week. Several of these students have been hired by LHS to write applications and systems programs and some, upon entering Berkeley as Freshmen, have joined the staff of the LHS Computer Education Project.

Similarly, OMSI's Community Research Center provides approximately 40 students per year an opportunity to carry out independent science research projects using the museum's PDP 11/45 computer. This is a more formal research training program for which there is a tuition fee, and in which more concentrated access to the computer laboratory and staff is provided. Here again, acceptance into the program is based on quality of a student-initiated proposal. In addition to high school credit, many of these students have earned awards and college scholarships as a result of their work.

Special Classes and School Field Trips. The remainder of the first level is comprised of organized groups of visiting school children. One large educational activity of all science-technology centers and of most museums is providing demonstrations, classes and field trip programs for school children. While these children represent an extremely large group compared to the 'special kids,' they are still a specially selected group, and their access to the institution, and thus to the computer, is strictly controlled and quite limited. Furthermore, at this level of the model, the computer is only available when a staff member of the institution is present to offer a scheduled class or workshop. Thus, even if children who learned something about the computer during a school field trip were to return to a 'Level I' institution on a weekend, for example, the computer would not necessarily be available for further individual exploration.

Programs and activities offered by science centers for visiting school children vary considerably, depending on the size of the institution, its staff, and especially on the computer facilities available. The California Museum of Science and Industry in Los Angeles, the Franklin Institute in Philadelphia, and the Fernbank Science Center in Atlanta are examples of institutions having restricted computer access as described above. The following three examples are taken from institutions at all levels so as to describe more adequately the range of computer activities available to visiting school children. Notice the dramatic difference in field trip programs at institutions further up the ladder from Level I.

At both the California Museum of Science and Industry in Los Angeles and the Franklin Institute in Philadelphia, one APL terminal connected to a remote IBM 360 or 370 (with the terminal and/or computer time subsidized by IBM) is used in group demonstrations of up to 50 children at a time. This is accomplished by seating the children amphitheater-style around the terminal and using a closed circuit television camera and monitor to give a close-up view of the action to the entire group. A museum staff member runs the demonstration, making use of as many volunteer student 'assistants' as possible. Using the APL language, the museum instructor demonstrates the machine's ability to do various arithmetic computations, formula evaluations, information storage and retrieval, a form letter and perhaps some games or typewriter pictures and posters[4]. (Similar demonstrations are offered to the public at a few announced times on weekends.)

The Franklin Institute also makes use of this amphitheater-style group presentation in its computer-based *Choices Theatre*. In this presentation, students are shown slides of various future scenarios and asked to vote for the most desirable alternative. The results of the voting are tabulated and immediately entered into a Hewlet Packard 9830 BASIC-speaking 'desktop computer' which acts as a scorekeeper and referee as well as speeding computation time during the presentation. Results of the voting and calculated statistical projections based on these results are displayed on two large TV monitors so that students can get a feeling for what might happen in the future if they make certain kinds of choices today.

The Children's Museum in Boston uses a different approach. Five CRT terminals (connected to its PDP 11/40 computer) and several Wang calculators are located in one corner of its visitor center. The CPU console, with its switch register (all switches disabled) and flashing lights also faces into this computer corner. Rather than attending formal demonstrations or workshops on computers, visiting school children are simply free to play with the calculators, press the CPU switches, and explore the various games offered on these terminals—at their own pace and on their own initiative[2].

The Lawrence Hall of Science, with its two computer classrooms, each containing 10 teletypes (connected to a Data General NOVA 800-based time-sharing system), combines the best of both of these approaches. Organized, staffed workshops are offered to groups of up to 30 visiting school children at a time. A brief introduction to computers is given to the visiting students by a staff member of the LHS Computer Education Project (these staff members are all college students at Berkeley and work part-time for the project). Then the students spend an hour, in groups of 2-3 per terminal, exploring a variety of games and simulations from the LHS Program Library while the LHS staff member circulates around the room, answering questions and helping students who have difficulties. Lawrence Hall also has computer terminals located in its exhibit area, so these students may interact further with computer terminals during any time that they are allowed free exploration of the exhibits.

To summarize, this is the nature of Level I, Special Access: either a very few special children learn a great deal about computers or a great many children have a very brief introduction, often with no avenue for follow-up. Because so little data are available on the current state of computing activities and science centers and museums, it is difficult to determine how many institutions besides Los Angeles, Philadelphia, and Atlanta would best be classified at this level. It is certain that Ground Level and Level I, together, best describe the current state of public access to computing at a majority of science centers and museums in the nation.

Slowing down for the curve. Students race against a computer-driven car during their visit to the Lawrence Hall of Science Computer Lab.

Part II of this article, which will appear in the next issue of *People's Computers*, will explore Level II of our Public Access Model. The model places more value on programs that encourage follow-up opportunities so that once people have become more interested in computers, they will have an opportunity to learn more about them and they will have a place where they can go to use them. Also, the model is based on access for the *general public* which includes adults as well as children, and a broad spectrum of people rather than selected groups. Level II shows a definite step upward in regard to both of these points.

References

1. Amara, Roy. *Toward Understanding the Social Impact of Computers.* Report R-29, Institute for the Future, 2740 Sand Hill Road, Menlo Park, CA 94025, May 1974.

2. 'Bursting at the Museums.' *Computer Decisions,* vol. 7, no. 3, March 1975, pp. 44-45.

3. Davis, Kay and Lee Kimche, ed. *Survey of Education Programs at Science-Technology Centers,* (ASTC), 2100 Pennsylvania Avenue, N.W., Washington, D.C. 20037, 1976. $4.50 incl. postage and handling.

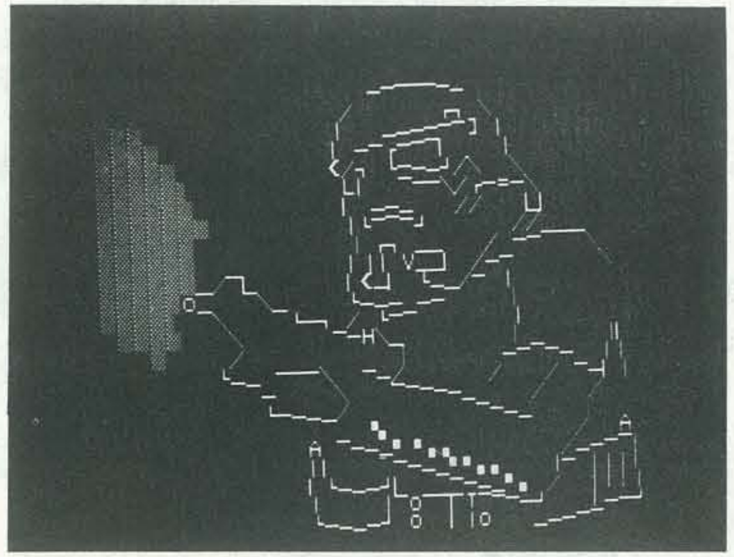
4. Goldwater, Daniel L. 'Games People Play-On Computers.' *Museum News,* vol. 51, no. 8, April 1973, pp. 30-34.

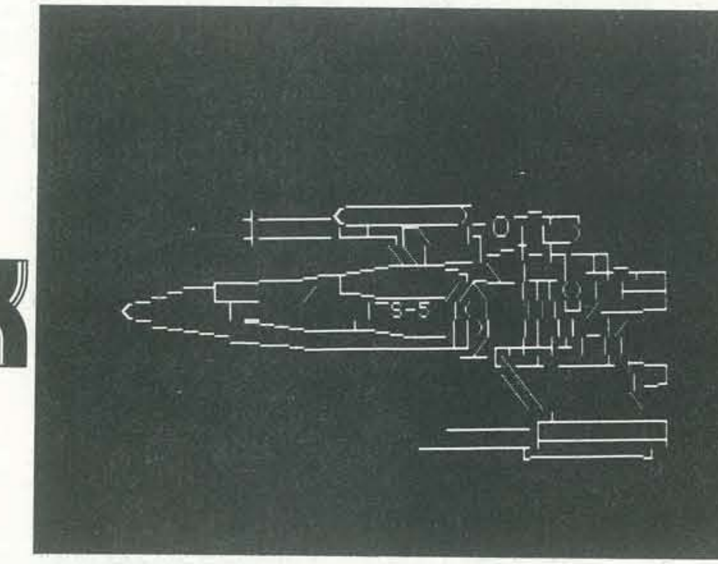
5. Kastner, Carol H. and William G. Underhill. 'Survey of Public Access to Computing.' *Proceedings of the National Computer Conference,* 1976, pp. 231-236.

6. Kemeny, John G. *Man and the Computer.* New York: Charles Scribner's Sons, 1972.

7. Kimche, Lee, ed. *Discover and Learn at Science-Technology Centers (A Guide to ASTC Science Centers).* ASTC, 2100 Pennsylvania Avenue, N.W., Washington, D.C. 20037, May 1976. $2.00 incl. postage and handling. □

ascii graphix





BY DANIEL BROWNING

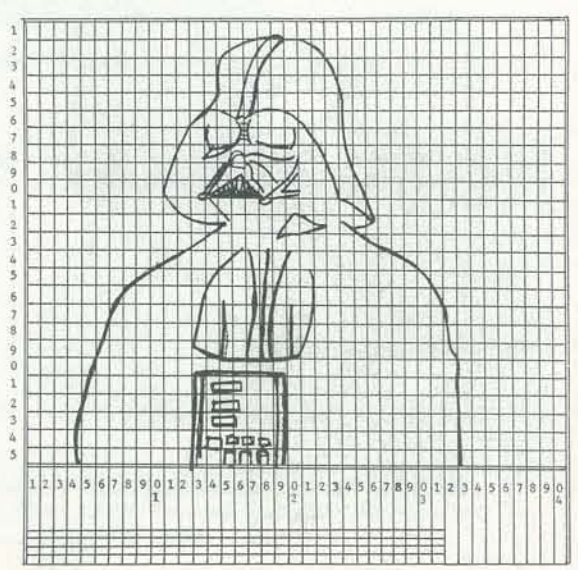
Not long ago we received this description of 'ASCII-Graphics' from Daniel Browning along with a cassette tape containing something called GRAPHIX-I and GRAPHIX-II. Upon loading this tape into a PET, we were delighted to discover four lovely examples of Dan's clever use of PET graphic symbols to create interesting pictures. Indeed, with Dan's permission, we have reproduced one of his scenes on the cover of this issue, and the others are reproduced here, accompanying Dan's explanation of how he makes them. –BK

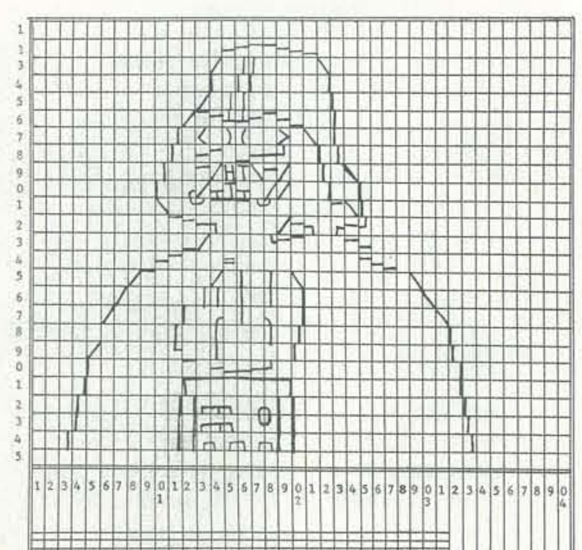
I am pleased to announce a fun way to do graphics on the Commodore PET, or any computer that has a 40 column and 25 line output. I call it the 'ASCII-Graph'. This involves a piece of graph paper specifically tailored to the PET's format. With this system, one can virtually reproduce photographs with ASCII characters, and of course, the PET's extensive graphic capabilities.

Here is how it works:

1. I first take a thin piece of paper, such as vellum, and place it over the desired photo.

2. I then trace the important outlines of the photo. (Preferably in ink.)





3. Now I place an ASCII-Graph over this outline, and retrace it onto the ASCII-graph, in pencil as shown in Figure 1.

4. In ink, I draw in the character that best fits the line which fills that particular box of the ASCII-Graph. This is illustrated in Figure 2. (Note that Figures 1 and 2 are drawn on the same ASCII-Graph. They are shown separately here for clarity.)

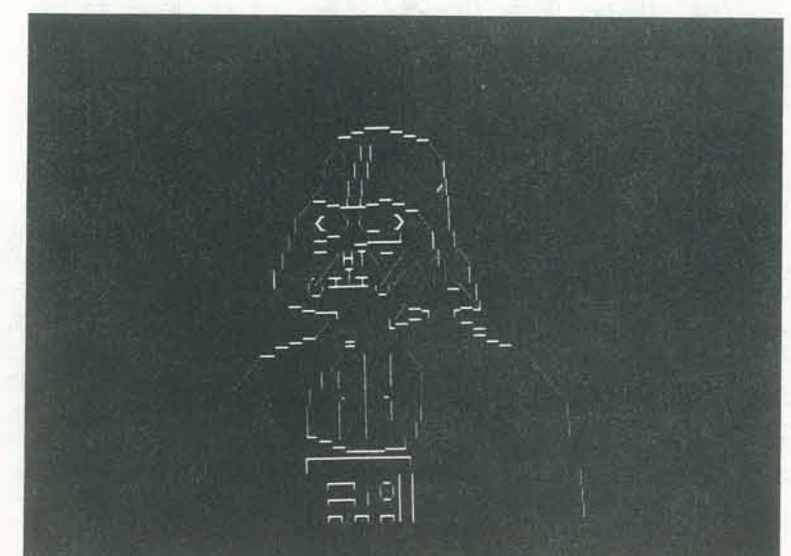
5. Type it in, using for example, TAB, SPC, or any other function that best suits your needs. Et voila!

Note: Because the ASCII-Graph is not transparent, both the original and the graph must be held up to a light source. I taped them to a window, one on top of the other, and easily penciled the image onto the graph.

I have also developed a new technique which requires no light source. The ASCII-Graph is now on acetate, not paper. Because acetate is transparent, it can simply be laid over the original. Instead of a pencil, a yellow felt-tip pen can be used to trace the original, and a black felt-tip can be used to fill in the characters . . . all on the same sheet.

The pictures shown are examples of my process. They were produced by GRAPHIX-I and GRAPHIX-II, two 4K PET programs. Readers interested in obtaining blank acetate ASCII-Graphs or software descriptions and a price list for the GRAPHIX programs illustrated, as well as some new ones, should send a self-addressed, stamped envelope to:

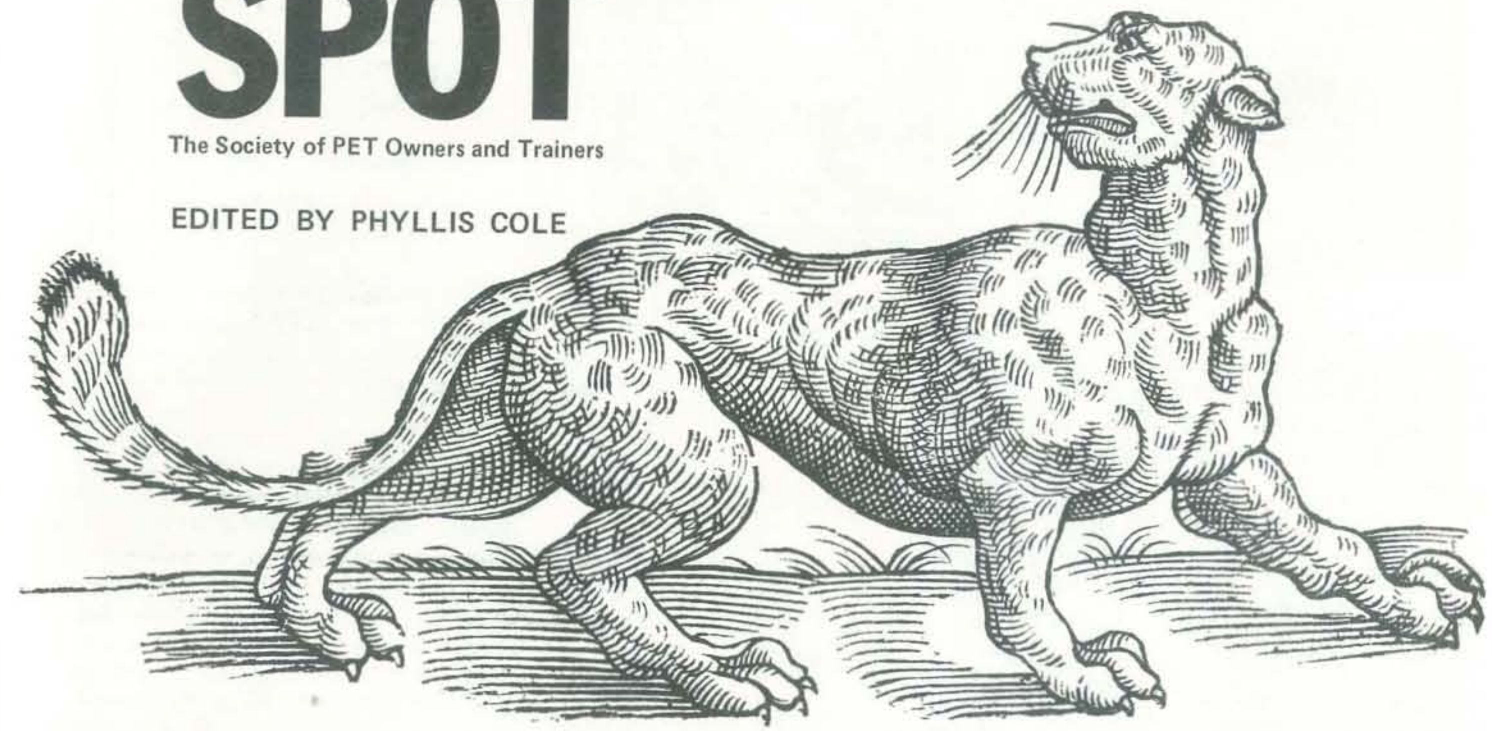
RAMA Electronics
1816 Lake St.
San Francisco, CA 94121 □



SPOT

The Society of PET Owners and Trainers

EDITED BY PHYLLIS COLE



Since I received my $795 PET home computer last October, I've been regularly reporting on it in the pages of People's Computers. *As more and more PETs enter the world, more and more readers submit programs, articles, and interesting information to share: please join the effort. Last issue we published names of some purveyors of PET software and newsletters as well as a number of graphics programs; this issue is quite different. Next issue we'll publish an article by Mark Zimmerman which includes tables of* decimal *op codes and mnemonics.*

—Phyllis Cole

ODD SPOTS

COMMODORE CORNER

Alas, Commodore has announced that its printer ($595, prints PET graphics, even lets you design one character yourself!) won't be available until *October*. And we'll have to wait until *November* for Commodore's floppy disk. OK entrepreneurs—start interfacing stuff to PETs and see if you can get a head start on Commodore—there are lots of us waiting for peripherals.

FREE PET SERVICES

Two free PET Computer services are available through the Microcomputer Resource Center.

1) The PET Cassette Exchange. Expand your program library easily. Exhange programs for the PET computer on cassette for free, no service charge. Simply send a tape of your programs and receive a tape with twice as many different programs on it. Please include enough stamps to have it returned to you via FIRST CLASS mail.

2) The Ultimate PET, a resource Handbook. A continually updated listing of all hardware and software sources for the PET. Send a self-addressed stamped envelope for your free copy.

Our Center has a PET and a MITE printer. We would be grateful for help in connecting the two. We then hope to offer a FREE service of printing out program listings: you would send us the cassette tape, we would list it and return it and the tape, free, no service charge. Also, we are hoping to get a modem and connect it to the PET. We are working on animated programs now. They will be available to anyone interested. Thanks.

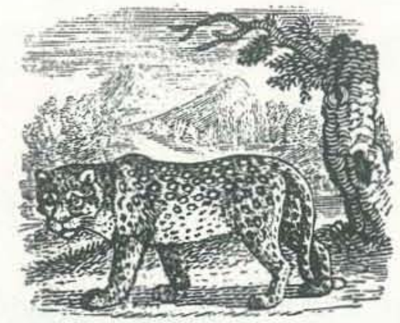
Mail your inquiries to Len Lindsay, Director, Microcomputer Resource Center, Inc, 1929 Northport Dr #6, Madison, WI 53704.

WRITE-PROTECTING TAPES

Cassette tapes may be protected from being accidentally erased. Locate the two small plastic fillers (or two thin plastic tabs) opposite the tape side of the cassette housing. Then, use a straightened-out paper clip to pry out these fillers; or use a ballpoint pen to bend the plastic tabs inward. Now your tape is 'write protected:' the RECORD button cannot be depressed when the cassette is in the recorder. To un-write-protect your tape, put a small piece of tape over the hole that's at the left-rear corner when the tape is in the recorder. After recording, you can remove the tape if you wish to write protect the new recording.

TAPE TALK

BY PHYLLIS COLE



In the last few months I've had many chances to talk to people about the problems of buying/exchanging PET programs, especially since I've been helping Peninsula School distribute PET software. Almost all PET owners seem to have experienced some difficulty LOADing a PET program. A small percentage of tapes may be defective, but in general, the problem is in the PET that is trying to LOAD.

If you are unable to LOAD even programs SAVEd on your own PET, your record/playback head probably needs cleaning and demagnetizing. See Commodore's instruction booklet.

Sometimes a program loads without a ?LOAD ERROR message but gets a ?SYNTAX ERROR message when it is RUN. This sometimes occurs when a single line of the program has been incorrectly LOADed. This situation can be corrected if you have a listing of the program you're trying to LOAD: compare the erroneous line with the listing, correct it, and SAVE the corrected version on another tape.

If you are able to LOAD programs that were SAVEd on some PETs but not programs that were SAVEd on other PETs, then the record/playback head on your PET is probably aligned differently from those on the other PETs. This situation is frustrating because it is hard to tell whether the fault is in your own machine or in the other one. If you send your PET back to the factory, you may be without it for several weeks, and after the alignment is changed, you may have difficulty LOADing programs you previously had SAVEd.

Aligning heads is difficult, and Commodore has not yet perfected their production techniques in this area. Very early PETs are especially likely to have misaligned heads. Commodore is quite aware of the problem because they are attempting to duplicate and sell software themselves.

If your PET cannot LOAD a program at all, some possible remedies are:

(1) Clean and demagnetize your heads, and try LOADing several times. WHEN YOU SUCCEED, SAVE THE PROGRAM ON ANOTHER TAPE.

(2) Find a friend with a PET and try to LOAD the program on that machine. When you succeed, SAVE on another tape and try to LOAD that tape on your machine. This works surprisingly often.

(3) Type the program in from a listing. Some people find that they learn a lot about BASIC from this exercise; other find the chore tedious.

(4) Have your playback head aligned.

In addition, those who have purchased software can

(5) Ask the manufacturer/supplier if you can get a tape recorded by a different method. Enclose a note explaining what problems were encountered with what programs, and the serial number of your PET and any other PETs you tried.

(6) Request a refund. Enclose a note detailing the problems and the serial number of you PET. The tape and documentation can be sent at 'Printed Matter' rates through the U.S. Postal Service.

Even if you are among those who have had no trouble LOADing PET tapes, we encourage you to support the efforts of everyone involved with PETs to make program interchange reliable.

PLOT

BY PHILIP GASH

Philip Gash has contributed a program which plots any single-valued function y(x) on a 50 by 80 grid. The user inserts the function into lines 510-550 of the program.

PLOT is great fun for graphing all sorts of functions from those studied in first year algebra on up. As you can see from the photos, the quarter-size rectangles are used to plot points. It's pretty easy to get weird results by choosing functions whose domain (i.e., x values) or range (y values) go beyond the graphic capabilities of the PET. But it's also easy and rewarding to get the types of results illustrated here.

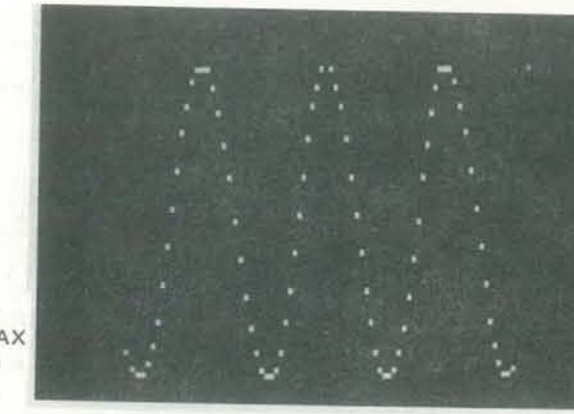
The photos were taken using Polaroid film type 107c in a camera with a special hood which positions the camera the proper distance from the screen and keeps glare and reflections out of the picture.

To use the program, first type the desired single-valued function into lines 510-550 of PLOT, then RUN it. You will be asked for the maximum and minimum values of x and y (see lines 600-625 of the program); then your function's graph will be displayed.

I've modified the two lines of the program submitted by Philip so that 'READY.' would not appear on the graph. Instead, line 160 GETs a character ZZ$ from the keyboard. If nothing is typed (i.e., ZZ$=""), then the program stays on the same line. This has the effect of simply causing the program to wait until some character is typed. When something is typed, the program goes on to line 165 which causes those lines containing the user function to be LISTed so that you can easily see what function you've been working with—and modify it, if you desire. —Phyllis Cole

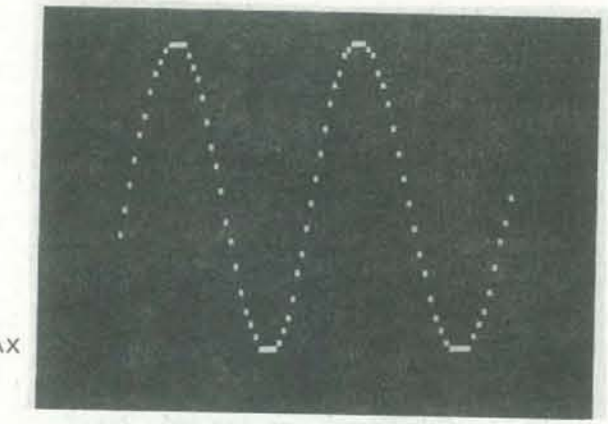
PLOT

```
  1 REM* PROGRAM PLOTS Y(X) WHICH IS *
  2 REM* TO BE INSERTED BY THE USER  *
  3 REM* IN LINES 510-550. PROGRAM   *
  4 REM* WRITTEN BY PHILIP GASH,     *
  5 REM* REDDING, CALIF. PROGRAM FOR *
  6 REM* INDIVIDUAL USE AND NOT FOR  *
  7 REM* SALE.                       *
  8 REM* PRINT "[CLR]"               *
  9 GOSUB 600
 10 FOR T=0 TO 7:READ CC(T):NEXT T
 12 DATA 98,127,226,255,123,126,108,124
 20 FOR P=0 TO 79 STEP 2
 25 X=P:GOSUB 490:Y1=Y:X1=X
 30 X=P+1:GOSUB 490:Y2=Y:X2=X
 35 M1=33728+INT(X1/2)-40*INT(Y1/2)
 40 M2=33728+INT(X2/2)-40*INT(Y2/2)
 45 P1=INT(Y1)-2*INT(Y1/2)
 50 P2=INT(Y2)-2*INT(Y2/2)
 55 IF M1<>M2 THEN 85
 60 IF M1<32768 OR M1>33767 GOTO 150
 65 PT=P1+P2
 70 IF P1=0 AND P2=1 THEN PT=3
 75 IF P1=1 AND P2=0 THEN PT=1
 80 POKE M1,CC(PT)
 82 GOTO 150
 85 IF M1<32768 OR M1>33767 GOTO 100
 90 PT=4
 92 IF P1=1 THEN PT=5
 95 POKE M1,CC(PT)
100 IF M2<32768 OR M2>33767 GOTO 150
105 PT=6
110 IF P2=1 THEN PT=7
120 POKE M2,CC(PT)
150 NEXT P
160 GET ZZ$:IF ZZ$="" GOTO 160
165 LIST 500-550
490 D=X:F=(X4-X3)*D/80+X3
495 X=F
500 REM* USER INSERTS Y=Y(X) ON *
505 REM* LINES 510-550          *
555 Y=(Y-Y3)*50/(Y4-Y3)
560 Y=D
570 RETURN
600 INPUT "MAX VALUE OF X";X4
610 INPUT "MIN VALUE OF X";X3
620 INPUT "MAX VALUE OF Y(X)";44
625 INPUT "MIN VALUE OF Y(X)";Y3
630 PRINT "[CLR]"
640 RETURN
```



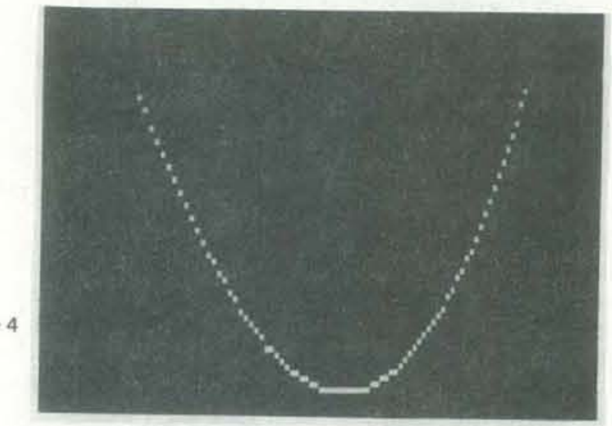
Y = COS(X)

	MIN	MAX
X	–10	10
Y	–1	1



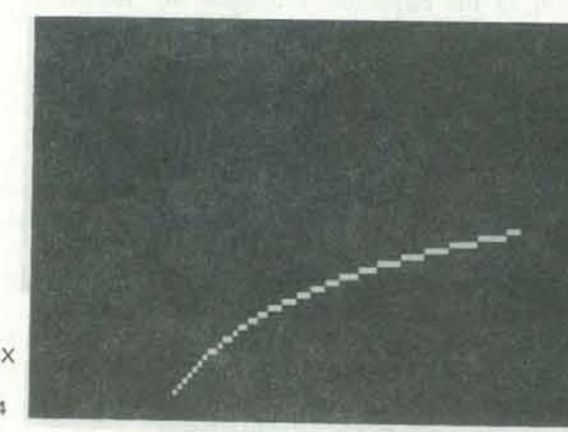
Y = SIN(X)

	MIN	MAX
X	0	13
Y	–1	1



Y = X ↑ 2 – 4 * X + 4

	MIN	MAX
X	0	4
Y	0	4



Y = LOG(X)

	MIN	MAX
X	.01	10
Y	.004	4.34

VIDEO MIXER

BY RANDALL JULIN

Although I have made design modifications to the PET Video Mixer circuit on the opposite page, I must give major credit where credit is due: the original circuit design was the work of Marc Hertzberg, Engineering Technician, and Ludwig Braun, Professor of Engineering, both of SUNY, Stony Brook, New York. Professor Braun is the Assistant Director for Educational Technology at SUNY, Stony Brook and is the author of 'Microcomputers and Video Disc Systems: Magic Lamps for Educators?', which has been printed in recent issues of computer journals, such as *Calculators/Computers* and *People's Computers*.

I have earned a Master's degree in Educational Technology at San Francisco State University, and I am now Manager of a computer-timesharing lab for the Information Science Program here on campus. For the last four or five years my interests have covered the spectrum of educational technology, and within the last two years or so I have become especially interested in the applications of computers in the classroom and in the educational curriculum. Because I see the potential of the *interactive* computer as a tool for learning in the classroom, I decided to buy a PET micro from Commodore last Christmas. Since then I have been trying to sharpen my PET BASIC programming skills; I have been learning a lot by attending the PET User Group meetings in Mountain View and by Exchanging PET programs on cassettes with other PET owners. I also have been able to pick up some very useful PET info and programs by subscribing to the *PET Paper* and the *PET User's Group Newsletter*, or by buying pertinent issues of the top popular computing magazines.

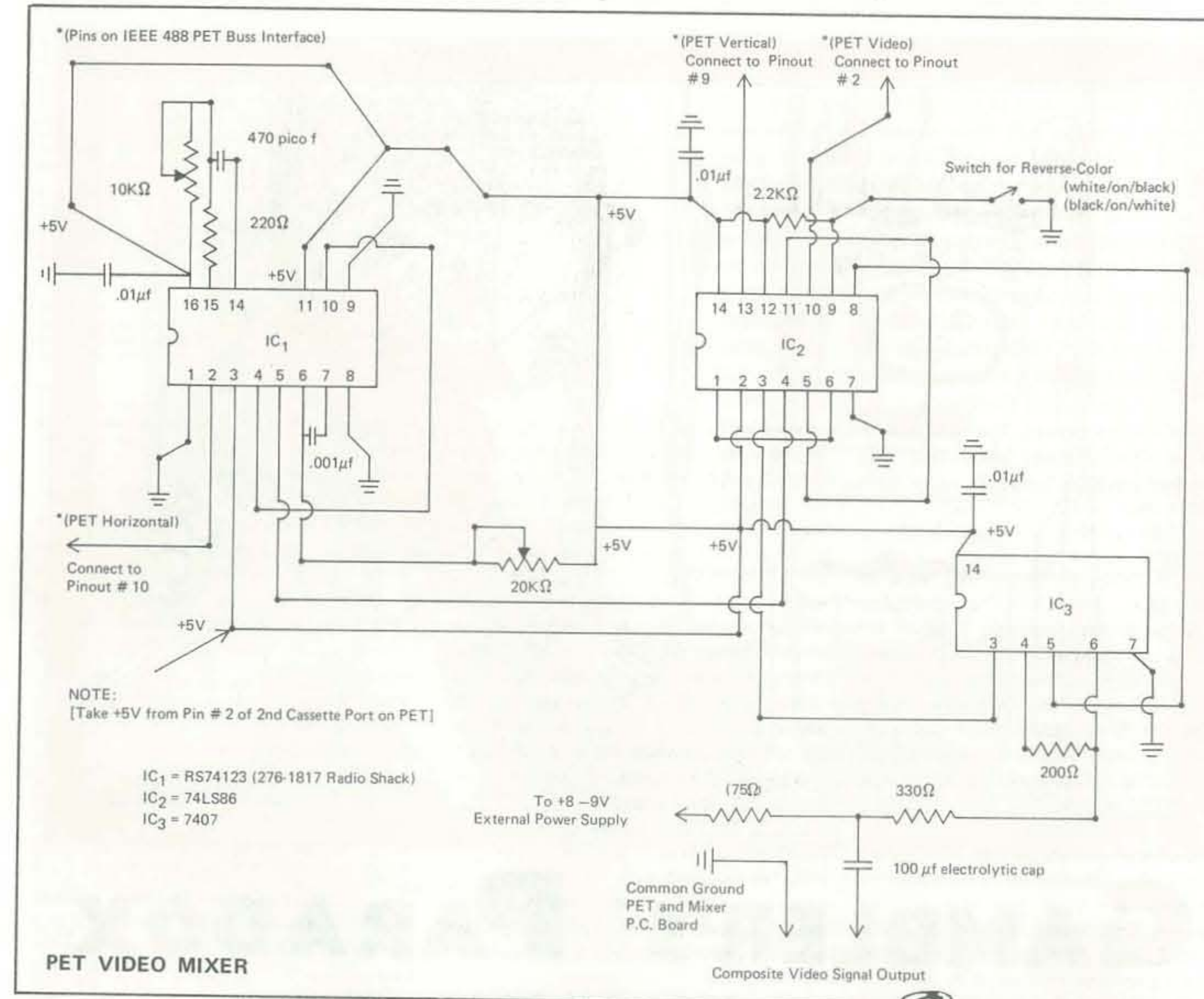
It was in *People's Computers* that I first read a letter from Professor Braun, in which he mentioned a circuit that would mix the three video signals put out by the PET IEEE 488-bus. This interested me very much, so I wrote a letter to Braun at Stony Brook, asking him to send me the details on such a circuit. Soon I had a circuit diagram for the PET Mixer and I went to work wiring up my version of the circuit, which required certain modifications. With the advice of Mike Butler, an engineering technician here at SFSU, I modified the circuit for an RS74123 IC chip (from Radio Shack, rather than another IC chip originally specified). The original circuit also called for a 12 volt positive supply voltage across a 100 microfarad electro-cap to boost the composite video signal output. I found that 12 volts was unnecessarily high to drive the video monitor I was using. I used a makeshift unregulated power supply of around 8 or 9 volts for this purpose.

I mounted the ICs on mounting plugs rather than hard-wiring them to the PC board. The components are mounted and wired in close to the ICs. It is a good idea to plan ahead in order to effectively use common ground lines and a common bus for the +5 volts. The PC board should be wired to interface with the PET's IEEE 488-bus, so the horizontal (pin 10), the vertical (pin 9) and the video signals (pin 2) can be inter-connected to the board easily. The +5 volts can be tapped off of pin 2 of the PET's second cassette port. There are two pots in this circuit which are to be used for adjusting the horizontal centering on the monitor screen. (I used a Panasonic CCTV video monitor with a 21 inch screen.) You might have to experiment with different output supply voltages, depending on what kind of monitor you are using.

The circuit for the PET mixer works beautifully (after I had taken care of a few cold solder joints and shorts). On several occasions I have used the mixer with a large screen monitor to demonstrate PET graphics programs and Peninsula School's 'LEMON' simulation to Information Science Lab visitors and groups of educators interested in the potential for using computers in school classrooms. A really neat feature, especially with graphics programs, is the ability to switch from black-on-white to white-on-black on the large monitor with the aid of the PET Video Mixer.

My thanks again to Professor Braun of SUNY, Stony Brook, and expertise of Marc Hertzberg (at SUNY) and Mike Butler (at SFSU). The potential for creative fun and learning with the PET micro used in a classroom environment or in a computer lab is limitless. Commodore has done educators a great service by putting its PET computer out on the market at a price which the average family can afford. It may take some time before many computers can be used in the classrooms as an integral part of the curriculum on any widespread basis, but in the meantime people are beginning to really appreciate the potential of the PET in their own homes. Applications of educational technology, and especially micros like the PET computer, might just be signalling the beginning of a true educational revolution.

Finally, I would like to applaud journals like *People's Computers*, and the maybe less-known, *MICRO: the 6502 Journal*, for providing forums for information regarding personal computers from the common people. Editors like Robert Tripp and Phyllis Cole should really be thanked for their help in letting us print what we have to say about personal computing in education today.



PET VIDEO MIXER

PETting A DIABLO

BUECK/JENKINS
2716 Stewart Lane
Rocky Mount, NC 27801

We have finally made our PET speak (in print) using a Diablo daisy wheel printer interfaced with Dick Rosner's PET ADA from Connecticut Micro Computer, 150 Pocono Road, Brookfield, Connecticut 06804.

The device converts IEEE 488 output to RS232 and works very well. Our initial start up didn't work because two pins on the Diablo (6 & 4) must be jumped. We also had trouble with tabbing, line feed, form feeds, etc. Diablo uses ESC and a character to perform these functions. To accomplish tabbing, use Print #5, CHR$(27); CHR$(09); CHR$(B) where B is the variable position on the page and CHR$(27) is ESC in decimal form and CHR$(09) is HT on horizontal tab. This sequence causes an absolute tab to the horizontal position on the line. By using both absolute vertical and horizontal tabbing, the print position can be quickly located anywhere on the page. The key to the typewriter control is to use CHR$ and decimal equivalent of the keytop sequence. Similar statements cause the Diablo to print in Red, Black, Sound Bell, Carriage Return, etc. We scratched our heads 'til a call to CMC put us on track.

Although the PET ADA may be a bit expensive @ $169 for a complete unit including case and power supply, the stripped for $98; it is our first usable short cut to print on a quality printer.

We hope Commodore will get moving on their own peripherals, but in the meantime, try the PET ADA. □

GAMBLER'S PARADOX

A LESSON IN COIN TOSSING USING COMPUTER SIMULATION

BY BOB KAHN

This article is actually a computer-based tutorial that I wrote awhile back to illustrate how a simple computer simulation game may be used as an experimental laboratory for exploring some elementary statistical concepts–in a gambling context. The text that follows is meant to be used in conjunction with a computer terminal (or a microcomputer) on which the two programs, FLIPME and GESTRA are available. For the purposes of this article, sample program runs have been inserted in the two places where one would normally go on-line.

Program listings are included at the end of the article. The FLIPME program may seem a little lengthy for a coin-tossing simulator; it is designed to stand alone as a game (independent of this tutorial) and contains a relatively elaborate set of messages which flip-flop from trial to trial depending on the outcomes. GESTRA is a simple simulation comparing four possible strategies that a human player might use in playing FLIPME.

This unit is designed as a set of experiments and computer programs dealing with the age old game of 'coin flipping' as a way to introduce you to some elementary concepts of statistics. We will use the computer as a handy coin simulator, calculating tool and record keeper to save a lot of time and also have some fun.

Let's try a very simple experiment–we have written a program that will enable the computer to play an exciting old game called FLIPME. The computer will simulate the flip of a *fair* coin and you will predict the results–H for Heads or T for Tails; each time you guess right, you will be given one morp (a morp is like a silver dollar in computer money). But alas, each time you are wrong, you lose a morp. You will start out a pauper with nothing to your name and will be given sixteen flips to improve your holdings. If you should end up in the hole (with a negative number of morps) you owe the computer . . . and it has a very good memory for such things!

O.K. Let's do it–load and run program FLIPME:


```
RUN
WHAT IS YOUR NAME?BOB KAHN
WOULD YOU LIKE INSTRUCTIONS?YES

HI BOB.  WE ARE GOING TO PLAY A GAME WITH MY SUPER
FLIP-FLOP COIN.  YOU START WITH  0  MORPS
[MORPS ARE LIKE SILVER DOLLARS].  THEN I WILL FLIP THE
COIN 16 TIMES AND EACH TIME YOU ARE TO GUESS HOW IT WILL
LAND -- H OR T. IF YOU ARE RIGHT, I WILL GIVE YOU ANOTHER
MORP; IF YOU ARE WRONG, HOWEVER, YOU LOSE 1.

O.K. BOB HERE GOES ...
YOU HAVE  0 MORPS.
FLIP  1    YOUR GUESS IS ?H
IT'S IN THE AIR ... AND IT'S ... TAILS
SORRY; YOU LOSE A MORP.
YOU NOW HAVE -1  MORPS.

FLIP  2    YOUR GUESS IS ?T
IT'S IN THE AIR ... AND IT'S ... TAILS
GOOD GUESS; YOU GET A MORP.
YOU NOW HAVE  0  MORPS.

FLIP  3    YOUR GUESS IS ?H
IT'S IN THE AIR ... AND IT'S ... HEADS
RIGHT AGAIN; I'LL PAY YOU ANOTHER.
YOU NOW HAVE  1  MORPS.

FLIP  4    YOUR GUESS IS ?T
IT'S IN THE AIR ... AND IT'S ... TAILS
BRING ME A NEW COIN, BOSS; THIS GUY'S TOO HOT!
YOU NOW HAVE  2  MORPS.
                .
                .
                .
FLIP  12   YOUR GUESS IS ?T
IT'S IN THE AIR ... AND IT'S ... TAILS
BRING ME A NEW COIN, BOSS; THIS GUY'S TOO HOT!
YOU NOW HAVE  0  MORPS.

FLIP  13   YOUR GUESS IS ?H
IT'S IN THE AIR ... AND IT'S ... TAILS
SORRY; YOU LOSE A MORP.
YOU NOW HAVE -1  MORPS.

FLIP  14   YOUR GUESS IS ?T
IT'S IN THE AIR ... AND IT'S ... HEADS
WRONG AGAIN; YOU LOSE ANOTHER.
YOU NOW HAVE -2  MORPS.

FLIP  15   YOUR GUESS IS ?H
IT'S IN THE AIR ... AND IT'S ... TAILS
ALAS, YOU LOSE STILL ANOTHER, BUCKWHEAT.
YOU NOW HAVE -3  MORPS.

FLIP  16   YOUR GUESS IS ?T
IT'S IN THE AIR ... AND IT'S ... TAILS
GOOD GUESS; YOU GET A MORP.
YOU NOW HAVE -2  MORPS.


AS THE SAYING GOES, 'YOU WIN A FEW, YOU LOSE A FEW.
AT THE RATE YOU'RE GOING, YOU'LL NEVER GET RICH--BUT
THEN YOU'LL NEVER STARVE EITHER.


LET'S SEE EXACTLY HOW YOU DID, BOB :

I FLIPPED THE COIN  16 TIMES:

THE RESULTS WERE  5  HEADS AND  11 TAILS.
YOU GUESSED HEADS  8  TIMES AND TAILS  8  TIMES,
AND YOU WERE CORRECT 7 TIMES, LEAVING YOU WITH -2  MORPS.

WOULD YOU LIKE TO TRY AGAIN, BOB?NO
ALL RIGHT; IT'S BEEN FUN PLAYING FLIPME WITH YOU.
NOW RETURN TO THE TEXT FOR A DISCUSSION OF THE GAME.
```

Welcome back. Presumably you have now played FLIPME at least once, and most likely you ended up with somewhere between two morps ahead and two morps behind (two morps represents one standard deviation in this game and there is actually about a 78% chance that your score was within one standard deviation above or below zero). If you did much better, perhaps you have extrasensory perception (ESP) or something–we can check out this possibility later. If you did much worse, you had best stay away from the gambling houses until your luck improves.

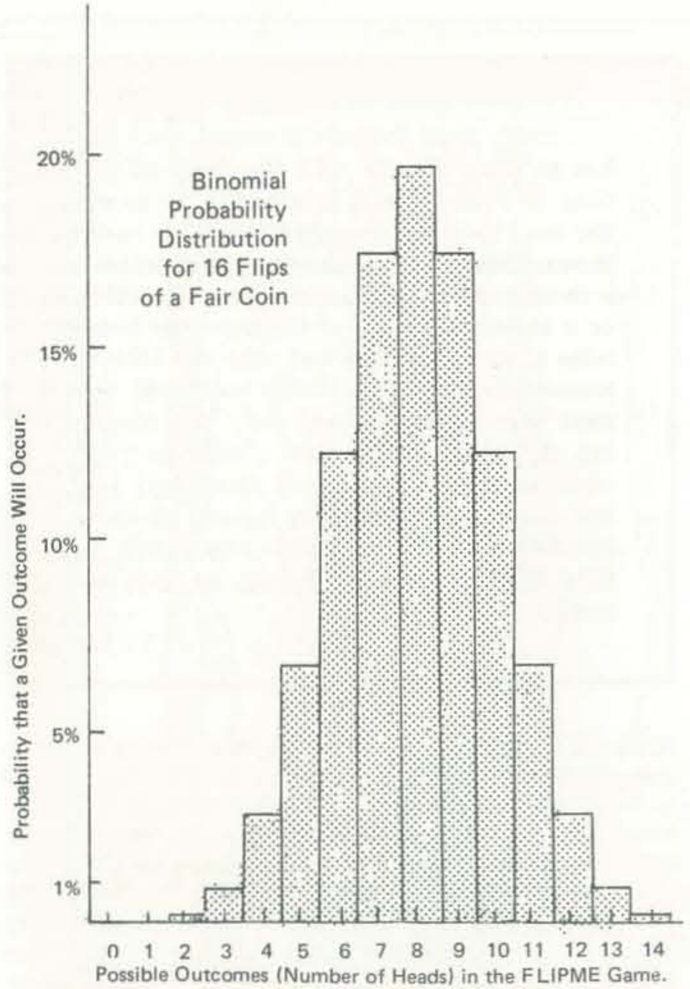
'. . . Every time the coin is tossed, each side has an equal chance with the other of winning. If heads wins it is just as likely to win the next time and the next and so on to the thousandth; so that on reasonable grounds a thousand heads in succession are possible, or a thousand tails; for the fact that heads wins at any toss does not raise the faintest reasonable probability that tails will win next time. Yet the facts defy this reasoning. Anyone who possesses a halfpenny and cares to toss it a hundred times may find the same side turning up several times in succession; but the total result will be fifty-fifty or as near thereto as does not matter. . .'*

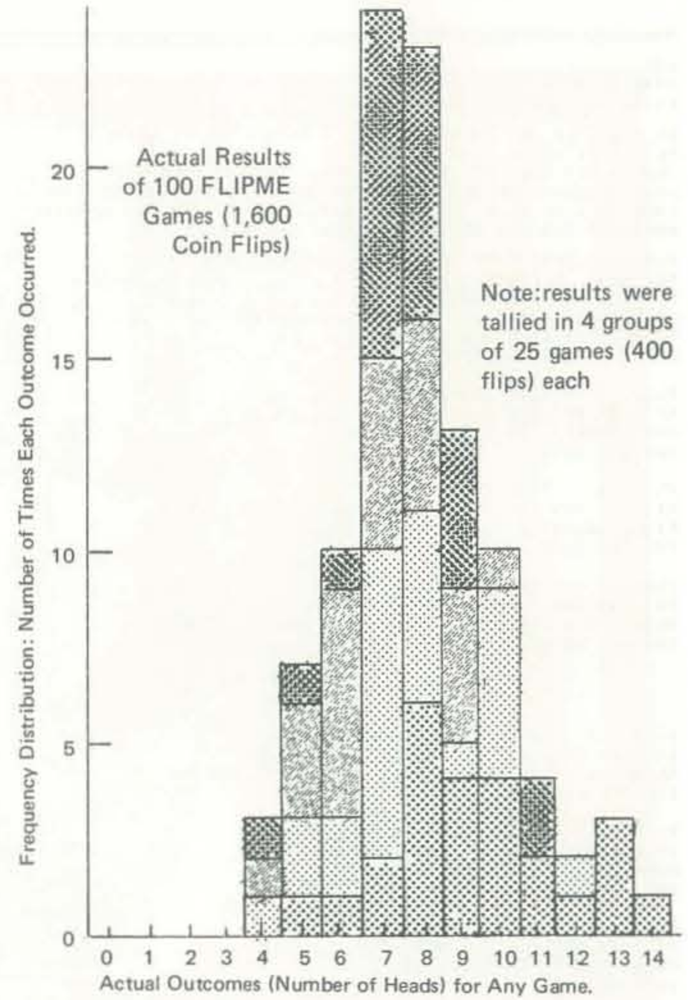
While playing the game FLIPME you were working on the *assumption* that our simulated coin was a fair one–that is, heads and tails should come up with approximately the same frequency. Keeping this assumption in mind, (we will later verify it for the computer coin) how many morps would you *expect* to gain *in the long run* by playing FLIPME? That is, let's say you played FLIPME 25 times (that's 400 flips); how many morps do you suppose you would end up with? When you have an answer to this question, go back and look at the results of your actual run of FLIPME. How many times did *you guess* heads? How many times did heads *actually* come up?

Reasoning from the fair coin idea we would *expect* heads to come up 50% of the time. Indeed, statisticians would say that .50 (which is the probability of a fair coin coming up heads) times the number of flips (16 in this case) gives us the *expected value* of heads . . . 8 out of every sixteen flips. Quite obviously, however, everyone who plays FLIPME will not obtain results of exactly 8 heads and 8 tails even with the fairest of coins (our computer simulated coin is just as 'fair' if not more so than a real coin). Some will find the coin coming up 7 heads and 9 tails; some 9 heads and 7 tails; some 11 and 5, and so on. The coin could even come up heads or tails *every time* although it is quite unlikely (one time in 65,536). However, in the *long run* if we look at the outcomes of all these FLIPME games, the *average* number of heads which turn up would be 8.

This does not completely answer our original question, though. Our expected gain in morps is also dependent on matching the guess with the actual outcome of the coin–and it is this possibility of a gambler's clairvoiance which is apparent in the lure of Las Vegas, Tahoe, Reno, Monte Carlo and other renowned gambling centers.

*From 'The Vice of Gambling and the Virtue of Insurance' by George Bernard Shaw as quoted in Marascuilo, L., *Statistical Methods for Behavioral Science Research*. New York: McGraw Hill, 1971.





We can see that a person who is guessing the results of a coin which has *already been flipped* has exactly two possibilities: (s)he is right or (s)he is wrong—regardless of what the coin came up. In FLIPME we have the reverse situation with the same end result; namely, what the person calls does not influence how the coin will land in any way. Thus, in either case, the expected gain of a person calling a fair coin should be .50 which is the probability of guessing right times the payoff for a correct guess minus .50 which is also the probability of guessing wrong times the penalty for a wrong guess. In FLIPME, this amounts to:

(.50 x 1 morp) + (.50 x -1 morp) = 0 morps

As you can see, the same results would be obtained if you were merely paid 1 morp each time the coin came up heads and docked 1 morp for each tail with no guessing at all! However, it must be emphasized that this is only true in the *long run* since a mere 16 flips is likely to yield any of several outcomes both favorable and not.

Despite all this logic and statistical theory, there is a very compelling psychological phenomena which leads most of us—at least sometimes—to doubt that different guessing strategies will lead to exactly the same results in the long run. This suggests another experiment: we will have the computer compare different guessing strategies on the same set of coin tosses to see if there are any significant differences as measured by net gain in morps. This time, we will have the computer do all the work to save time and fingers. We have chosen four strategies as representative of all the countless possibilities; they are as follows:

1. Always guess heads
2. Guess Heads and Tails alternating—H T H T etc.
3. Start with Heads; guess the same outcome until you are wrong, then switch to the other alternative and so on
4. Guess at random with no set rule or pattern

The program is set up so that you may choose the number of flips in the experiment and also you may choose whether or not you wish to see all the outcomes of the strategies for each flip.

The first time through the program, ask for 16 flips (the same as in FLIPME) and have the program list all the outcomes so you can see how the strategies compare on each flip. Then, when the program asks if you wish to repeat the experiment say 'YES' and give it 400 flips but do not have it list them or you will be there for half an hour. Actually, we need at least 400 flips to assure ourselves that our super-computer-simulated coin is fair. Statistically, 400 flips enables us to say that the coin is fair with 95% certainty if it comes up heads anywhere from 180 to 220 times. So far, our computer-simulated coin has been well within this range.

The program which performs this simulated comparison is called GESTRA. Now, you should load and run this program just as you did FLIPME:

```
GET-GESTRA
RUN

THIS PROGRAM WILL COMPARE FOUR GUESSING STRATEGIES
IN TRYING TO MATCH ANY NUMBER OF TOSSES OF THE SUPER-
FLIP-FLOP COIN.  THIS IS A SIMULATION USING PREDETERMINED
DECISION RULES--NOTHING HAS BEEN RIGGED OR FIXED.
YOU MAY SEE ALL THE OUTCOMES IF YOU DESIRE.


HOW MANY TIMES SHOULD I FLIP THE COIN ?16
WOULD YOU LIKE TO SEE THE OUTCOMES ?YES

TRIAL   COIN      S1        S2        S3        S4

 1       H         H         H         H         T
 2       T         H         T         H         T
 3       T         H         H         T         T
 4       T         H         T         T         H
 5       T         H         H         T         T
 6       H         H         T         T         T
 7       T         H         H         H         H
 8       H         H         T         T         T
 9       T         H         H         H         T
 10      T         H         T         T         T
 11      T         H         H         T         H
 12      H         H         T         T         T
 13      T         H         H         H         T
 14      T         H         T         T         T
 15      H         H         H         T         H
 16      T         H         T         H         H

                    STRATEGIES
      S1 - ALWAYS GUESS HEADS
      S2 - GUESS HEADS AND TAILS ALTERNATELY
      S3 - GUESS PREVIOUS OUTCOME TILL WRONG, THEN SWITCH
      S4 - GUESS RANDOMLY


THE RESULTS WERE  5  HEADS AND  11  TAILS.

                              S1    S2    S3    S4

NUMBER OF HEADS GUESSED       16    8     6     5
NUMBER OF MATCHES             5     7     7     8
NET OUTCOME IN MORPS         -6    -2    -2     0

DO YOU HAVE ANOTHER PROBLEM ?YES

HOW MANY TIMES SHOULD I FLIP THE COIN ?400
WOULD YOU LIKE TO SEE THE OUTCOMES ?NO


                    STRATEGIES
      S1 - ALWAYS GUESS HEADS
      S2 - GUESS HEADS AND TAILS ALTERNATELY
      S3 - GUESS PREVIOUS OUTCOME TILL WRONG, THEN SWITCH
      S4 - GUESS RANDOMLY


THE RESULTS WERE  196  HEADS AND  204  TAILS.

                              S1    S2    S3    S4

NUMBER OF HEADS GUESSED       400   200   196   204
NUMBER OF MATCHES             196   208   206   206
NET OUTCOME IN MORPS         -8     16    12    12

DO YOU HAVE ANOTHER PROBLEM ?YES
```

You have probably run through GESTRA at least twice now. Notice that each strategy might have a different net outcome (in morps) but that on the average, you end up with none. This is much less likely to be the case with 16 flips than with 400 or more. By the same token, you may be alarmed to have gotten a net outcome as high as 26 (or as low as -26) out of 400 flips from one strategy while another gave an outcome of 2. Although these results might lead you to believe that one strategy is superior to the other, these are relatively small deviations from the expected net outcome of 0 morps—especially when compared to the number of flips (400). However, such a conclusion would not be justified on the basis of 16 flips since even very small deviations from 0 would be large proportionally. Statisticians use a technique called hypothesis testing to examine such outcomes statistically for significance at given levels of error tolerance. As an example, the 400-flip run of GESTRA shown below turns out to have *no* significant differences in the outcomes of the four strategies with the probability of error in making such a decision minimized to 5%. You will notice, however, that the strategies definitiely give somewhat different payoffs in this case, as the table shows.

```
HOW MANY TIMES SHOULD I FLIP THE COIN ?400
WOULD YOU LIKE TO SEE THE OUTCOMES ?NO


                    STRATEGIES
      S1 - ALWAYS GUESS HEADS
      S2 - GUESS HEADS AND TAILS ALTERNATELY
      S3 - GUESS PREVIOUS OUTCOME TILL WRONG, THEN SWITCH
      S4 - GUESS RANDOMLY


THE RESULTS WERE  182  HEADS AND  218  TAILS.

                              S1    S2    S3    S4

NUMBER OF HEADS GUESSED       400   200   183   187
NUMBER OF MATCHES             182   206   199   203
NET OUTCOME IN MORPS         -36    12   -2     6

DO YOU HAVE ANOTHER PROBLEM ?YES
```

Thus, we must conclude that no matter how we guess in the coin tossing game, the expected outcome will be based on a 50-50 chance of guessing correctly and if we play the game many times, we will lose as much as we gain.

In another unit we will discuss hypothesis testing in more detail and we will generate what is known as a confidence interval for the probability of our coin coming up heads based on a set of trial observations. This is the beginning of a set of tests to determine whether or not our computer coin is really 'fair' or if it is actually biased. We will then purposely bias the coin to see if any of our guessing strategies will pay off with a biased coin even though they did not matter with a fair one.

FLIPME

```
LIST
   1 REM -FLIPME:  COIN FLIPPING GAME
   2 REM - PROGRAM BY BOB KAHN, REVISED 6/76
   3 REM - LAWRENCE HALL OF SCIENCE "DECISION" BASIC
   4 REM - (DATA GENERAL NOVA 800/ECLIPSE COMPUTERS)
   5 REM - UNIVERSITY OF CALIFORNIA, BERKELEY
  20 DIM N$[20],F$[10],G$[5],A$[5]
  30 DIM I$[10]
  40 PRINT "WHAT IS YOUR NAME";
  50 INPUT N$
  55 PRINT
  60 LET M=0
  70 LET R=16
  80 GOSUB 1010
  90 PRINT "WOULD YOU LIKE INSTRUCTIONS";
 100 INPUT I$
 105 PRINT
 110 IF I$[1,1]="N" THEN 140
 115 REM - IF I$ = YES TEHN PRINT INSTRUCTIONS; THEN INITIALIZE
 120 GOSUB 1060
 130 GOSUB 930
 140 LET G=0,H=0,S=0
 150 LET E1=2,E2=2
 160 PRINT
 165 REM - START GAME
 170 PRINT "O.K. ";F$;" HERE GOES ..."
 180 PRINT "YOU HAVE ";M;"MORPS."
 190 FOR K=1 TO R
 200   PRINT "FLIP ";K;
 210   PRINT "   YOUR GUESS IS ";
 220   INPUT G$
 225   PRINT
 230   IF G$[1,1]="H" THEN 280
 240   IF G$[1,1]="T" THEN 280
 250   PRINT "THIS COIN ONLY HAS TWO SIDES--HEADS AND TAILS; ANYTHING"

 260   PRINT "ELSE YOU TYPE WILL REPEAT THIS MESSAGE."
 270   GOTO 210
 275   REM - CALL COIN FLIPPING ROUTINE; F=0=TAILS; F=1=HEADS
 280   GOSUB 800
 290   ON F+1 GOSUB 880,910
 300   PRINT A$
 305   REM - COMPARE GUESS TO FLIP; PRINT OUT APPROPRIATE MESSAGE
 310   IF G$[1,1]=A$[1,1] THEN 390
 320   LET E2=2
 325   REM - E1 COUNTS THE NUMBER OF MATCHES (WINS) IN A ROW, MOD 3
 330   LET E1=E1+1-(INT((E1+1)/3)*3)
 340   ON E1+1 GOSUB 1430,1450,1470
 350   LET M=M-1
 360   PRINT "YOU NOW HAVE ";M;" MORPS."
 370   PRINT
 380   GOTO 450
 390   LET E1=2
 395   REM - E2 COUNTS THE NUMBER OF NON-MATCHES (LOSSES) IN A ROW,
 396   REM - MOD 3(
 400   LET E2=E2+1-(INT((E2+1)/3)*3)
 410   ON E2+1 GOSUB 1490,1510,1530
 420   LET S=S+1
 430   LET M=M+1
 440   GOTO 360
 450   IF G$[1,1]="T" THEN 470
 460   LET G=G+1
 470 NEXT K
 480 PRINT
 485 REM - S1 IS USED TO DECIDE WHICH FINAL WORDS OF WISDOM TO PRINT
 486 REM - BEFORE THE TALLY
 490 LET S1=INT(ABS(M)/4)*SGN(M)+5
 500 ON S1 GOSUB 1180,1210,1210,1240,1270,1310,1350,1350,1390
 510 PRINT
 520 PRINT
 530 GOSUB 1100
 670 PRINT "WOULD YOU LIKE TO TRY AGAIN, ";F$;
 680 INPUT I$
 685 PRINT
 690 IF I$[1,1]="Y" THEN 140
 700 PRINT "ALL RIGHT; IT'S BEEN FUN PLAYING FLIPME WITH YOU."
 710 PRINT "NOW RETURN TO THE TEXT FOR A DISCUSSION OF THE GAME."
 720 PRINT
 730 PRINT "TEAR OFF AND KEEP THIS PRINTOUT FOR FUTURE REFERENCE."
 740 PRINT
 750 PRINT "........................................................"
 760 FOR K=1 TO 8
 770   PRINT
 780 NEXT K
 790 END
 795 REM - FLIPPING ROUTINE
 800 PRINT "IT'S IN THE AIR ...";
 820 PRINT " AND IT'S ... ";
 825 REM - THE PAUSE STATEMENT BELOW IS FOR "DRAMATIC" EFFECT...
 840 PAUSE 15
 850 LET C=INT(RND(0)*10000)
 855 LET F=0
 860 IF INT(C/2)=C/2 THEN  LET F=1
 870 RETURN
 880 LET A$="HEADS"
 890 LET H=H+1
 900 RETURN
 910 LET A$="TAILS"
 920 RETURN
 930 PRINT "HI ";F$;".  WE ARE GOING TO PLAY A GAME WITH MY SUPER"
 940 PRINT "FLIP-FLOP COIN.  YOU START WITH ";M;" MORPS"
 950 PRINT "[MORPS ARE LIKE SILVER DOLLARS].  THEN I WILL FLIP THE"
 960 PRINT "COIN ";R;" TIMES AND EACH TIME YOU ARE TO GUESS HOW IT WIL
L"
 970 PRINT "LAND -- H OR T.  IF YOU ARE RIGHT, I WILL GIVE YOU ANOTHER
"
 980 PRINT "MORP; IF YOU ARE WRONG, HOWEVER, YOU LOSE 1."
 990 PRINT
1000 RETURN
1010 FOR Z=1 TO LEN(N$)
1020   IF N$[Z,Z]=" " THEN 1050
1030   LET F$[Z,Z]=N$[Z,Z]
1040 NEXT Z
1050 RETURN
1055 REM - ROUTINE TO PRINT BLANK LINES
1060 FOR Z=1 TO 3
1070   PRINT
1080 NEXT Z
1090 RETURN
1095 REM - ROUTINE TO PRINT THE RESULTS
1100 PRINT "LET'S SEE EXACTLY HOW YOU DID, ";F$;" :"
1110 PRINT
1120 PRINT "I FLIPPED THE COIN ";R;"TIMES:"
1130 PRINT
1140 PRINT "THE RESULTS WERE ";H;" HEADS AND ";R-H;"TAILS."
1150 PRINT "YOU GUESSED HEADS ";G;" TIMES AND TAILS ";R-G;" TIMES,"
1160 PRINT "AND YOU WERE CORRECT ";S;" TIMES, LEAVING YOU WITH ";
1165 PRINT M;" MORPS."
1166 PRINT
1170 RETURN
1175 REM - ROUTINES TO PRINT APPROPRIATE MESSAGES FOR VARIOUS OUTCOMES

1180 PRINT "YOU MUST HAVE SOME SORT OF REVERSE ESP--YOU WERE WRONG"
1190 PRINT "EVERY SINGLE TIME ..... INTERESTING!"
1200 RETURN
1210 PRINT "I'VE HEARD OF GAMBLER'S RUIN, BUT THIS IS RIDICULOUS."
1220 PRINT "GO HAVE A BEER AND TRY YOUR LUCK SOME OTHER TIME."
1230 RETURN
1240 PRINT "HAD A LITTLE STROKE OF BAD LUCK, EH?  WELL, YOU'LL"
1250 PRINT "SOON GET A CHANCE TO GAIN IT ALL BACK ... HE HE HE"
1260 RETURN
1270 PRINT "AS THE SAYING GOES, 'YOU WIN A FEW, YOU LOSE A FEW.'"
1280 PRINT "AT THE RATE YOU'RE GOING, YOU'LL NEVER GET RICH--BUT"
1290 PRINT "THEN YOU'LL NEVER STARVE EITHER."
1300 RETURN
1310 PRINT "YOU HIT A STREAK OF GOOD LUCK, EH--A SILVER LINING"
1320 PRINT "SO TO SPEAK; IF I WERE A SLOT MACHINE YOU'D PUT IT"
1330 PRINT "ALL RIGHT BACK, I'LL BET ...."
1340 RETURN
1350 PRINT "HOT DAMN! LAS VEGAS, TAHOE, RENO, MONTE CARLO, LOOK"
1360 PRINT "OUT!!  THIS KID IS REALLY HOT STUFF--YOU MUST HAVE X-RAY"
1370 PRINT "VISION OR SOMETHING."
1380 RETURN
1390 PRINT "YOU MUST HAVE ESP OR YOU PEEKED AT MY RANDOM NUMBER"
1400 PRINT "GENERATOR OR SOMETHING; NOBODY GET'S THEM ALL RIGHT WHEN"
1410 PRINT "I'M FLIPPING!"
1420 RETURN
1430 PRINT "SORRY; YOU LOSE A MORP."
1440 RETURN
1450 PRINT "WRONG AGAIN; YOU LOSE ANOTHER."
1460 RETURN
1470 PRINT "ALAS, YOU LOSE STILL ANOTHER, BUCKWHEAT."
1480 RETURN
1490 PRINT "GOOD GUESS; YOU GET A MORP."
1500 RETURN
1510 PRINT "RIGHT AGAIN; I'LL PAY YOU ANOTHER."
1520 RETURN
1530 PRINT "BRING ME A NEW COIN, BOSS; THIS GUY'S TOO HOT!"
1540 RETURN
```

GESTRA

```
LIST
   5 REM - GESTRA:  PROGRAM TO COMPARE 4 GUESSING STRATEGIES IN THE
  10 REM - COIN TOSSING MODEL (FAIR COIN)
  20 REM - PROGRAM BY BOB KAHN, REVISED 6/76
  30 REM - LAWRENCE HALL OF SCIENCE "DECISION" BASIC (DATA GENERAL
  35 REM - NOVA 800/ ECLIPSE COMPUTERS)
  40 REM - UNIVERSITY OF CALIFORNIA, BERKELEY
  45 DIM G[4],H[4],S[4],M[4],A$[5]
  50 PRINT
  60 PRINT "THIS PROGRAM WILL COMPARE FOUR GUESSING STRATEGIES "
  70 PRINT "IN TRYING TO MATCH ANY NUMBER OF TOSSES OF THE SUPER-"
  80 PRINT "FLIP-FLOP COIN.  THIS IS A SIMULATION USING PREDETERMINED"

  90 PRINT "DECISION RULES--NOTHING HAS BEEN RIGGED OR FIXED."
 100 PRINT "YOU MAY SEE ALL THE OUTCOMES IF YOU DESIRE."
 110 PRINT
 120 PRINT
 125 REM - INPUT NUMBER OF TRIALS; INITIALIZE & SET COUNTERS TO ZERO
 130 LET L=0,B=0,N=0
 140 PRINT "HOW MANY TIMES SHOULD I FLIP THE COIN ";
 150 INPUT K1
 155 PRINT
 160 FOR K=1 TO 4
 170   LET G[K]=0,H[K]=0,S[K]=0
 180 NEXT K
 190 PRINT "WOULD YOU LIKE TO SEE THE OUTCOMES ";
 200 INPUT A$
 205 PRINT
 210 PRINT
 220 IF A$[1,1]<>"Y" THEN 250
 230 LET L=1
 240 PRINT "TRIAL   COIN       S1        S2        S3        S4"
 250 LET A$="HT"
  55 REM - SIMULATE THE COIN TOSSING K1 TIMES
 260 PRINT
 270 FOR K=1 TO K1
 280   LET F=INT(2*RND(0))
 285   REM - F IS THE VALUE OF THE FLIP: F=0=TAILS; F=1=HEADS
 290   REM - N COUNTS THE NUMBER OF TIMES HEADS COMES UP
 295   REM - AFTER INCREMENTING HEADS COUNTER, APPLY STRATEGIES
 300   IF F=1 THEN 320
 310   LET N=N+1
 320   GOSUB 670
 330   GOSUB 730
 340   GOSUB 840
 350   GOSUB 930
 355   REM - L IS A FLAG; IF L=1 THEN LIST ALL OUTCOMES
 360   IF L=0 THEN 430
 370   PRINT K; TAB (10);A$[F+1,F+1];"     ";
 380   FOR T=1 TO 4
 390     LET P=G[T]+1
 400     PRINT "     ";A$[P,P];"    ";
 410   NEXT T
 420   PRINT
 430 NEXT K
 435 REM - WHEN ALL FLIPS COMPLETED, PRINT OUT SUMMARY RESULTS
 440 PRINT
 450 PRINT  TAB (25);"STRATEGIES"
 460 PRINT "      S1 = ALWAYS GUESS HEADS"
 470 PRINT "      S2 = GUESS HEADS AND TAILS ALTERNATELY"
 480 PRINT "      S3 = GUESS PREVIOUS OUTCOME TILL WRONG, THEN SWITCH"
 490 PRINT "      S4 = GUESS RANDOMLY"
 500 PRINT
 510 PRINT
 520 PRINT "THE RESULTS WERE ";N;" HEADS AND ";K1-N;" TAILS."
 530 PRINT
 540 PRINT  TAB (30);" S1    S2    S3    S4  "
 550 PRINT
 560 PRINT "NUMBER OF HEADS GUESSED"; TAB (30);H[1]; TAB (36);H[2];
 565 PRINT  TAB (42);H[3]; TAB (48);H[4]
 570 PRINT "NUMBER OF MATCHES"; TAB (30);S[1]; TAB (36);S[2];
 575 PRINT  TAB (42);S[3]; TAB (48);S[4]
 580 FOR T=1 TO 4
 590   LET M[T]=2*S[T]-K1
 600 NEXT T
 610 PRINT "NET OUTCOME IN MORPS"; TAB (30);M[1]; TAB (36);M[2];
 615 PRINT  TAB (42);M[3]; TAB (48);M[4]
 620 PRINT
 625 PRINT "DO YOU HAVE ANOTHER PROBLEM ";
 626 INPUT A$
 627 IF A$[1,1]="Y" THEN 110
 630 FOR T=1 TO 10
 640   PRINT
 650 NEXT T
 660 GOTO 1010
 670 REM-STRATEGY 1 -- ALL HEADS
 680 LET H[1]=H[1]+1
 690 IF F=1 THEN 710
 700 LET S[1]=S[1]+1
 710 LET G[1]=0
 720 RETURN
 730 REM-STRATEGY 2 -- HALF AND HALF
 740 IF INT(K/2)=K/2 THEN 800
 750 IF F=1 THEN 770
 760 LET S[2]=S[2]+1
 770 LET G[2]=0
 780 LET H[2]=H[2]+1
 790 RETURN
 800 IF F=0 THEN 820
 810 LET S[2]=S[2]+1
 820 LET G[2]=1
 830 RETURN
 840 REM- STRATEGY 3 -- GUESS PREVIOUS OUTCOME TILL WRONG THEN
 850 REM-CHANGE
 860 LET G[3]=B
 870 IF F<>B THEN 890
 880 LET S[3]=S[3]+1
 890 LET B=F
 900 IF G[3]=1 THEN 920
 910 LET H[3]=H[3]+1
 920 RETURN
 930 REM-STRATEGY 5 -- GUESS RANDOMLY
 940 LET G[4]=INT(2*RND(0))
 960 IF G[4]=1 THEN 980
 970 LET H[4]=H[4]+1
 980 IF G[4]<>F THEN 1000
 990 LET S[4]=S[4]+1
1000 RETURN
1010 END
```

CONCENTRATION

BY MILAN D. CHEPKO

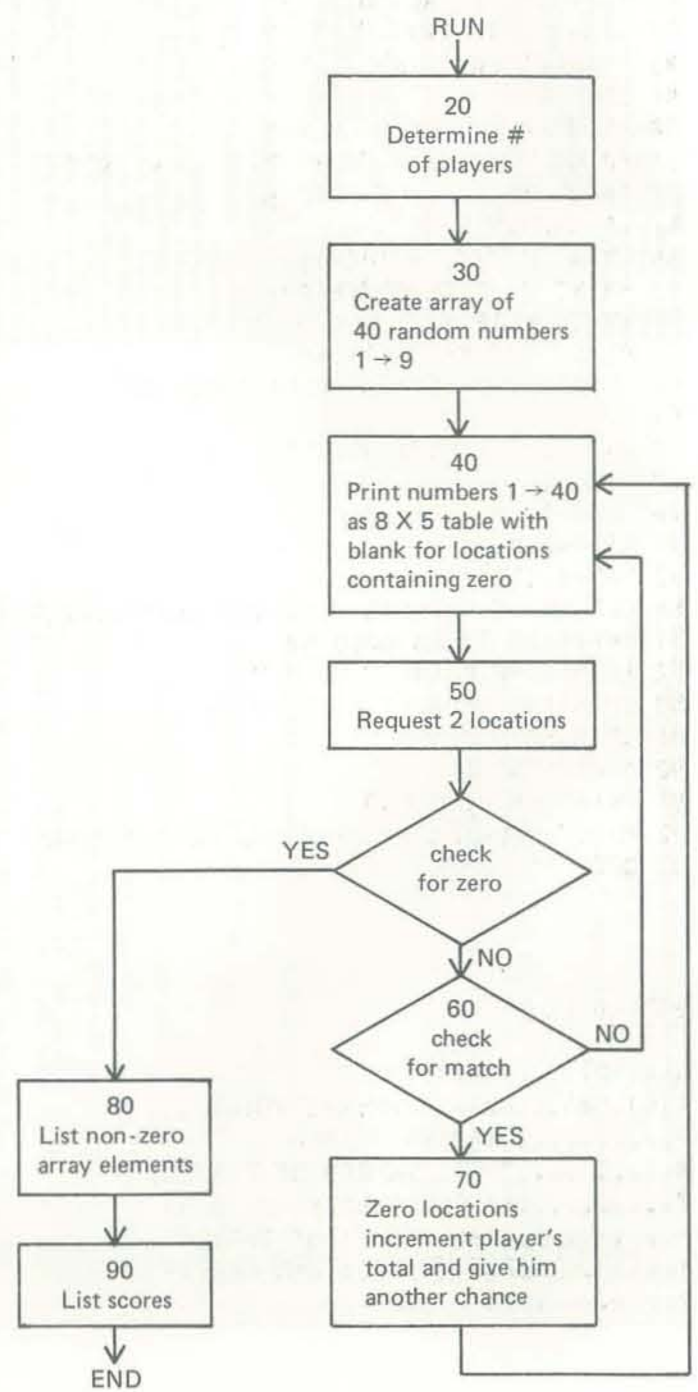
Milan Chepko is a modern-day Walter Mitty. In the real world, he maintains a medical practice in Thief River Falls, Minnesota. He apparently also has a 'secret' life in which he sits, hunched over his home-brewed, 8080 micro, turning out nifty game programs in Denver Tiny BASIC. . . 'pocketa, pocketa, queep, pocketa, pocketa, pocketa. . .'

Last issue, we published Milan's Tiny Blackjack program, and this month, we are pleased to publish his version of the old playing card (and TV quiz show) game, Concentration. Since we don't have a home-brewed 8080 micro handy right now, we quickly converted Milan's program to run on a TRS-80; thus, the listing shown in this article is our TRS-80 version of Milan's Denver Tiny BASIC program. –BK

The following is a short program that allows from 2–10 people to play the old card game of Concentration. In its original form, the game is played with a deck of 52 cards, which are shuffled and dealt face-down to cover a table top. One player turns over 2 cards and, if they match, removes them. If they don't match, they are turned face-down again and the next player selects 2 cards. At the end of the game, the player with the most cards wins.

The program generates an array of 40 elements, each containing a randomly chosen number from 1 to 9. (String variables could also be used to include the face cards J, Q, K, A.) Before each play, a grid of the numbers from 1 to 40 is printed out in an 8-column by 5-row pattern on the CRT; this could be modified for use on a teletype to save time and paper. A player then inputs two locations, and the contents of these are revealed . . . if they match, those locations are zeroed and the player is given credit for the match. When printing the grid during subsequent plays, any location that contains zero is displayed as a blank, as if the 'card' had been picked up from the table.

To end the game, a player will input a zero for the first location. The program will then display the contents of the unmatched locations and a listing of the point totals for each of the players.



CONCENTRATION

```
1Ø CLS:P=Ø:
11 PRINT "  ***CONCENTRATION***":PRINT
2Ø PRINT "HOW MANY PLAYERS (MAXIMUM=1Ø)";
21 I=Ø:INPUT N:IF N>1Ø GOTO 2Ø
22 I=I+1:IF I>N GOTO 3Ø
23 A(4Ø+I)=Ø:GOTO 22
3Ø I=Ø
31 I=I+1:IF I>4Ø GOTO 4Ø
32 Z=RND(9)
33 A(I)=Z:GOTO 31
4Ø CLS:I=Ø
41 I=I+1:IF I≥4Ø GOTO 5Ø
42 IF A(I)=Ø PRINT "  ", " ";
43 IF A(I)>Ø PRINT I, " ";
44 GOTO 41
5Ø PRINT:P=P+1:IF P>N P=1
51 PRINT "PLAYER # ";P;" PICK A NUMBER";
52 INPUT X:IF X>4Ø GOTO 51
53 IF X=Ø GOTO 8Ø
54 PRINT "#";X;"=";A(X);
55 PRINT "  PICK ANOTHER";
56 INPUT Y:IF Y>4Ø GOTO 55
57 PRINT "#";Y;"=";A(Y)
6Ø IF A(X)>Ø IF A(X)=A(Y) GOTO 7Ø
61 J=Ø
62 J=J+1:IF J>3ØØ GOTO 4Ø
63 GOTO 62
7Ø A(4Ø+P)=A(4Ø+P)+1
71 A(X)=Ø:A(Y)=Ø
72 P=P-1:GOTO 4Ø
8Ø CLS:PRINT "HERE'S WHAT WAS LEFT":I=Ø:PRINT
81 I=I+1:IF I>4Ø GOTO 9Ø
82 IF A(I)=Ø PRINT " ", " ";
83 IF A(I)>Ø PRINT A(I), " ";
84 GOTO 81
9Ø I=Ø:PRINT
91 I=I+1:IF I>N END
92 PRINT "PLAYER # ";I;" YOU HAVE ";A(4Ø+I);" POINTS"
93 GOTO 91
```

20-23	Sets point total for each player = 0
30-33	Sets up 5 x 8 array of 40 random numbers from 1 to 9
40-44	Prints sequential 5 x 8 array of numbers 1 to 40. Prints a *blank* if an element contains zero instead of a number from 1 to 9
50-57	Each player in turn selects 2 locations in the array, and the contents are displayed. If 0 is entered for the first location, the game ends
60-63	Checks for a match. If none, enters a timing loop. Note that only *non-zero* elements qualify!
70-72	On a match, increments player's total and zeroes those two elements. (P=P-1 allows that player to have a second turn)
80-90	At end of game, prints the remaining non-zero elements
91-93	Lists each players point total

VARIABLES

```
A(I:4Ø)....ARRAY
A(41:5Ø)...EACH PLAYER'S POINT COUNT
P..........CURRENT PLAYER
N..........TOTAL NUMBER OF PLAYERS
Z..........RANDOM NUMBER (1 TO 9)
X..........PLAYER'S FIRST CHOICE
Y..........PLAYER'S SECOND CHOICE
I,J........COUNTERS
```

□

DRAGONSMOKE

BY THE DRAGON

The Dragon (alias Bob Albrecht) started this magazine way back in 1972. For its first four years, when it was an oversized, funky newspaper called PCC, he was also the editor. Indeed, you will find part of The Dragon's tale (heh, heh) in The Modern-day Medicine Show article on p.6 of this issue. Recently, The Dragon has become interested in setting up and running fantasy games for children. Dragonsmoke is his forum for sharing information and ideas in this regard. –BK

F & SF GAMES CONVENTIONS

Got the computer faire blahs? Next time, stretch your mind in a new direction–try a Fantasy and Science Fiction Games Convention.

PACIFIC ENCOUNTERS * * * SEPTEMBER 2-4, 1978. VILLA HOTEL, SAN MATEO, CALIFORNIA.

- *Live*, role-playing fantasy adventure games with play time ranging from 5 hours to 24 hours (for you robots and alien creatures who need no sleep).
- Computer Games.
- Fantasy and Science Fiction board games.

For info, contact: Pacific Encounters, P.O. Box 4042, Foster City CA 94404 or call (415) 523-6747.

GEN CON XI * * * AUGUST 17-20, 1978, PARKSIDE CAMPUS, KENOSHA, WI

- Tournaments
- General Gaming
- Exhibits
- Auctions
- Seminars
- Movies
- Miniatures
- Science Fiction
- Fantasy

SPECIAL EVENT!
DUNGEON CHAMPIONSHIP
Info: GEN CON XI, P.O.B. 756, Lake Geneva, WI 53147, (414) 248-3625.

COSMIC ENCOUNTER

I just received a copy of the board game *Cosmic Encounter* and haven't had time to play it yet. Will do–and report on it in a future 'Dragonsmoke.' In the meantime, here are excerpts from a review in the May 1978 issue of *THE DRAGON* (TSR Hobbies, P.O.B. 110, Lake Geneva, WI 53147).

"Innovation of any sort is always something of a risk, and that axiom holds true even in the wargame market. The risk can be magnified if the producer of the game is not one of the hobby's stalwarts, but a new company proffering its initial endeavor.

But then, there are times when the newcomer really shines. Eon Product's *Cosmic Encounter* is one of those instances.

If you read any of the many SF magazines or gaming journals, you have probably already encountered an ad for this game. It has probably been one of the most widely advertised SF games to date. I know that when I first saw the ad I was curious and a bit skeptical about the game; perhaps it would be a good game, but would it appeal to SF gamers of a serious sort?

When I received my copy, I opened it with eager anticipation of what type of game *Cosmic Encounter* would be. I found some people to play it with, quickly read the rules (they are only a few pages long) and sat down to play the game. I was quite impressed. We played two games that first night and twice that many on the following day.

It is hard to say just what type of game *Cosmic Encounter* is; it defies the normal categories we wargamers deal with. It is certainly not a standard style game. There are no factors on the counters, no CRT or associative dice, and movement is not regulated by the familiar hexgrid. Rather, the game is a combination of RISK-style strategic games and role playing games, with a diplomatic action thrown in. The result is highly playable, fanciful, and very fun.

Cosmic Encounter plays quickly and simply, but its simplicity is misleading. Strategies are present, and must be followed to win. Players must be observant and know the potentials of his enemies as well as themselves. The game is something like *Go* in that play is much more thoughtful than a reading of the rules would imply.

Cosmic Encounter is billed as 'a science fiction game for everyone' and this is very true. The simplicity of the rules, short time of play (usually about 45 minutes) and the game's color make it appealing to non-wargamers. More serious players can enjoy its laid back atmosphere and simply have some fun. Both types of players will be rewarded with some very good times.

Cosmic Encounter is a new type of SF game (I should emphasize that last word as in no way is this a simulation or does it appear to strive to be) and it hits its mark quite squarely. From both a physical and design point of view it is a very good game."

Cosmic Encounter is available from Eon Products, Inc., 96 Stockton St., Dorchester, Mass. 02124 for $10.00 (boxed).

Reviewed by Tony Watson

THE SPACE GAMER

I *do* like the magazine called *The Space Gamer*. The Jan-Feb 1978 issue has articles on *Wizard* (a role-playing fantasy game) and *Olympica* (a tactical science fiction game); 'Byte the Bullet' (how to computerize a game based on Conway's *Life* with *two* kinds of beasties); how to put robots into your F & SF games; a review of *Imperium*, a science fiction game which combines strategy and tactical combat with politics and economics.
Info: Metagaming, Box 15346, Austin, TX 78761.

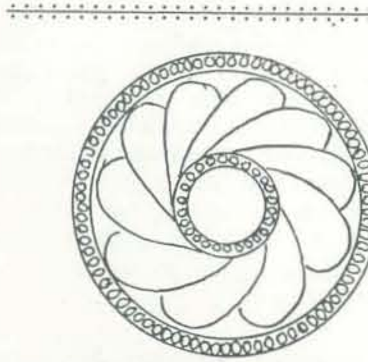
Watch DRAGONSMOKE for more pointy claws pointing toward sources of info on Fantasy and Science Fiction Gaming. Let the poor Klingons rest for awhile and play some *interesting* games! □

REVIEWS

CALCULATORS, COMPUTERS, AND ELEMENTARY EDUCATION
by David Moursund
Available from: Math Learning Center, 325 13th Street N.E., Unit 302, Salem, Oregon 97301
174 pp. $7.00

As an elementary school teacher who has used calculators and computers with children for several years I found this book informative and helpful. It is clear and easy to read and understand. I found the sections about pocket calculators particularly useful. I would recommend this book to others in the educational field interested in calculators and computers.

Reviewed by Dorothy M. Calabrese
Peninsula School
Menlo Park, CA 94025



THE CHEAP VIDEO COOKBOOK
by Don Lancaster
Howard W. Samms & Co Inc
256 pp, $5.95

A good cookbook is one which stirs you into action. When you read it, you immediately want to try out the recipes it contains. The ingredients should be readily available, and it helps if they are economical. The *Cheap Video Cookbook*, by Don Lancaster, has all these qualities.

I have put together several computer kits in the past but have never been stimulated to do much in the way of external hardware. That is, until I picked up Lancaster's latest cookbook. Not being initially enthusiastic about reviewing the book, the preface convinced me to investigate further. It sounded simple–five chapters, each clearly defined so as to make understanding easy and straightforward. The ingredients needed were few and inexpensive. The result promised to be an exotic enhancement (the TVT 6 5/8) to my newly acquired KIM microcomputer.

Chapter 1–Some Basics. An overall picture of the cheap video approach is gained in this chapter. The idea is to get you from your microcomputer to your TV set with a minimum of hassle. Quoting from the text, '. . . The method combines an absolute minimum of dedicated hardware with some operating software commands, and two new concepts called *upstream tap* and a *Scan microinstruction*. . . . (these) let you display 2000 upper and lower case ASCII characters on a largely or totally unmodified TV set *with stock video bandwidth*, at a total cost of around $20, and using only seven integrated circuits worth of dedicated hardware on a small *single-sided* PC card.'

If that quote doesn't 'turn you on', forget the book. However, I'm hooked now and must read on.

A 256 X 256 graphics display is another possibility mentioned, and full color graphics with somewhat lower resolution is said to be accomplished just as easily.

The philosophy behind it all is contained in five *first* principles to cheap video:

'1. Leave the existing microcomputer system nearly as you found it. . .
2. Use a plain old TV set, also leaving it nearly as you found it. . .
3. Put some hardware between the microcomputer and the TV set that lets them talk to each other.

4. Add two key elements to the microprocessor architecture: a *Scan microinstruction* and an *upstream tap*.
5. Use software and firmware sesequences to control what the interface hardware is going to do.'

The key to achieving success with this method is the interface which performs two important functions:

1. It causes the microcomputer to access its display memory in just the right way.
2. It delivers composite video that contains both bandwidth-enhanced characters and properly positioned horizontal and vertical synchronizing pulses.

The three key parts—microprocessor, interface and TV set (or video monitor)—are shown in a block diagram along with a description of their individual functions. A commercial version of the TVT 6 5/8 is described and photographs are shown of the product and examples of the video display produced.

Chapter 1 closes with an explanation of the basic theory and some design restrictions. The systems described were designed and debugged around the MOS Technology 6502 microprocessor and the KIM-1 and KIM-2 microcomputers. The author states that conversion to 6800 systems would not be too difficult, and the 8080 or Z-80 conversions are possible.

Chapter 2—Software Design. Whenever an interface is used to connect your computer to some outside device, there will be some trade-off between software and hardware. Given two interfaces which perform the same function, the one with less expensive, simpler hardware will require more complex software to drive it. Therefore, one might expect the software design of the cheap video to be rather complicated. Despite this complexity, the chapter is highly readable and will provide excellent reference on video techniques. And if you are going to use the TVT 6 5/8 on a KIM, 'all you will need are the results of this chapter without having to go into the gory details of where the results came from.'

In all, this chapter contains 75 pages of clear, concise instructions on the software design used for the cheap video with excellent practical examples.

Chapter 3—Hardware Design. A description of the interface card hardware design opens the third chapter accompanied by a block diagram. Then each of the interface hardware blocks is clearly and thoroughly presented. These include:

1. the instruction decoder
2. the Scan microprogram generator
3. high frequency timing circuitry
4. sync and position circuitry
5. bandwidth compensation and video output
6. computer interface
7. television interface

Once again, specific applications are given for the KIM. As in all chapters, a profusion of circuit diagrams and charts are provided to clarify the text. The chapter concludes with helpful suggestions for adapting an ordinary TV set for use with your computer.

Chapter 4—Building the TVT 6 5/8. This is the hardware buff's delight. From parts lists through schematic diagrams and PC boards patterns, a complete step-by-step construction procedure is clearly defined. How it works and what to do if it doesn't is also carefully explained. Complete construction details for display modules are given. These include:

1. Upper- and Lower-Case Alphanumeric Module
2. 256 X 256 Black and White Graphics Module
3. 96 X 128 Color Graphics Module
4. Upper Case Only Alphanumeric Module

One section of this chapter is devoted to debug and checkout procedures. Once again, it is a step-by-step approach and seems to be very complete. This feature is usually lacking in most build-it-yourself project documentation.

Chapter 5—Transparency. The book winds down with a discussion of methods to let your microprocessor do other things at the same time that you are displaying, or at least seem to do so.

The principles of transparency are discussed with the accompanying problems in timing and synchronization. Several methods of solution are suggested with sample flowcharts, diagrams and programs. Future enhancements to your cheap video system complete the chapter.

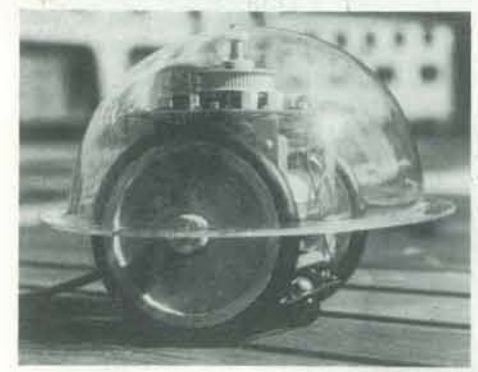
I feel that Lancaster has really outdone himself with the *Cheap Video Cookbook.* I have read raves of his other books, but this one tops them all. It provides a delicate balance of theory and practical solutions and leaves the reader with a desire to put the theory to work in a practical project. Charts, illustrations and diagrams contribute to understanding the text instead of detracting from it. The text is clear even though the theory gets complicated at times.

Any useful instructional material must make use of a specific model (or models) in order to be practical. Don has chosen the 6502 microprocessor and the KIM microcomputer. The book leans heavily on them to provide substance to his practical approach. Without such a model, the material would have to deal in generalities and would be much less useful. The 6502 makes an excellent model for the cheap video project.

Reviewed by Don Inman

ANNOUNCEMENTS

HARDWARE



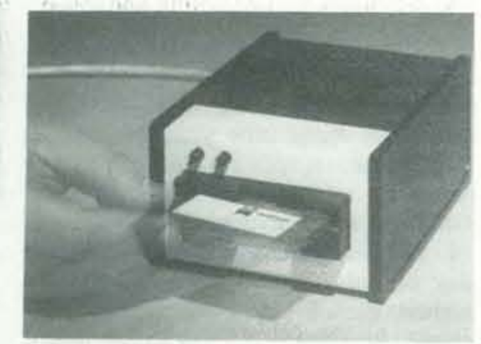
COMPUTER-CONTROLLED TURTLE

The Terrapin™ Turtle, a small electronic robot controllable by micro-processor can 'walk' (roll), touch (with its 3-1/2" radius hemispherical dome), and draw (lowering its pen attachment), as programmed. It has lights for eyes and a speaker to emit sounds. The Turtle requires a parallel interface: one compatible with an S-100 bus is available as an accessory. Each Turtle comes with ten feet of cable and may be purchased either as a kit or fully assembled. Each kit comes with a tested, 20-page instruction manual.

The Turtle may be used to map rooms, solve mazes, teach simple geometry or programming concepts, as well as many other tasks. The Turtle is 5" high, has a 3-1/2" radius, crawls at 6" per second and is extremely versatile due to its touch sensors. Brochures are available. Kit $300; assembled $500; interface $40. For further information contact: David McClees, President, Terrapin, Inc, 33 Edinborough Street, Sixth floor, Boston, MA 02111; (617) 482-1033.

FLOPPY TAPE WAFER FOR THE S-100

Exatron Corporation of Sunnyvale, California, has announced the Exatron Stringy Floppy™, an innovative mass storage subsystem for computers with the S-100 bus. The subsystem consists of a control board connecting to the bus, a small drive module outside the computer mainframe, and the connecting cable.



The individual continuous-loop tape wafer, less than a fourth the bulk of the standard audio cassette and holding up to 40K bytes, is inserted in the slot in the front of the drive module, and will save or load 4K bytes in about 5 seconds. All operations are software controlled; the utility programs are contained in an EPROM on the control board. The subsystem is assembled and tested at the factory, is delivered ready to operate, and is backed by a one-year full warranty and a 30-day money-back guarantee. An owners' association has been organized, monthly newsletters are being published, and weekly workshops will be conducted at the manufacturer's plant in Sunnyvale, on Saturday mornings for owners and prospective users.

For further information, contact EXATRON, 1030 E Duane, Suite 1, Sunnyvale, CA 94086; (408) 737-7111.

MICROAGE MKB-2 KEYBOARD

The new MKB-2 Keyboard by MicroAge, designed for use with the new 64 and 80 character display video boards, combines the most popular keyboard features with a low affordable price.

Included as standard in the MKB-2 are a numeric key pad, upper and lower case, cursor control keys, 2-key rollover, and auto repeat on all keys. Plus, the MKB-2 is assembled in a heavy duty steel case with parallel interface, strobe or pulse, on-board regulation (5v, 12v), complete with standard DB25S connector, and black double-injection molded keys. List: $149.00. Available from MicroAge, 1425 W 12th Place #101, Tempe, AZ 85281; (602) 967-1421.

540 VIDEO DISPLAY

Ohio Scientific is now offering the video display interface from its popular Challenger IIP as a fully assembled accessory for any OSI system. The 540 video display has also been incorporated in the company's mainframe class personal computer, the C2-8P and in two floppy disk based computer systems. The display features a 32 row by 64 column display of the standard 64 character ASCII font in 5 x 7 dot matrix form. Standard features include programmable formatting of the display for 32 x 32 or 32 x 64. The 32 x 32 mode is useful for video animation since it provides square character cells. The video board also features a keyboard port which can be used with a standard ASCII keyboard or OSI's new programmable keyboard. The 540 also optionally supports a graphics character generator which features lower case and about 170 special characters for plotting and gaming.

All systems using the 540 incorporate OSI's 542 programmed keyboard, which is a fully programmable keyboard system capable of upper and lower case and auto repeat on all characters. The keyboard also allows many special single keystroke commands and direct single keystroke graphics, as well as providing for character editing and special video game formats.

The model 540 video board is available as an add-on option for any existing OSI system as a CA-11 and retails for $249.00. The graphics character generator option will be available in June 1978, and will retail for $29.00. For more information, contact Ohio Scientific, 1333 South Chillicothe Rd, Aurora, OH 44202; (216) 562-3101.

TRANSIENT VOLTAGE SUPPRESSOR

Lightning and heavy-duty electrical equipment often create power-line surges and transients. These can cause extensive damage to valuable microporcessors and peripherals.

Electronic Specialists is announcing a line-cord transient suppressor which will absorb repeated power surges, protecting delicate equipment.

Available in 2 prong plug/socket ($11.50) or 3 prong plug/socket ($14.50), these units are also available with integral power line hash filtering.

ELECTRONIC SPECIALISTS, Box 122, Natick, MA 01760; (617) 655-1532.

NCE MAIL ORDER AUCTION SALE

Here's a rare opportunity to name your own price while purchasing some of the most famous names in Mini and Micro computers sold in America today. It's Newman Computer Exchange's annual summer mail order auction sale. All year long, NCE purchases lot quantities of both used and surplus new equipment from leading manufacturers and large volume users.

These items are reconditioned to qualify for Newman's standard 90 day warranty, and then sold through a variety of mail order catalogs. 'When an item doesn't move, or if it's overstocked for any reason, we set it aside for our annual summer auction,' says Chuck Newman, vice president of the firm. 'We establish minimum bids, usually below our cost, and then take our chances in a true auction market. If we receive any bid above the minimum, then the item is sold, even if it means we take a loss.'

This year's auction includes lot quantities of such best-selling items as the Commodore PET computer. Even on these 'best sellers,' minimum bids are well below list prices. Last year's auction catalog listed merchandise valued at over a quarter of a million dollars. The original retail value of the items listed this year will exceed $1,000,000.

Only bids received by mail will be accepted. Persons appearing in person on the day of the auction cannot out-bid the highest mailed bid which has been received. In addition to catalogs distributed at the NCC, in Anaheim, eighty thousand auction catalogs will be mailed early in June to Newman Computer Exchange mailing list members. The opening date for bids is set for Monday, July 31, 1978.

You can obtain your copy of the auction catalog and become a mailing list member by writing to: NEWMAN COMPUTER EXCHANGE, Department R59, P.O. Box 8610, Ann Arbor, MI 48107.

NEW BYTE SHOPPER AVAILABLE

The new Spring, 1978, edition of the BYTE SHOPPER, a guide to the world of personal computing, is now available through MicroAge. In keeping with microcomputer industry trends, the new edition provides the reader with a complete key to computer system selection, and a guide to disk drive selection. New sections focus on systems designed specifically for business applications.

Recognizing the need for education, the BYTE SHOPPER is also an introductory text to personal computing, providing a glossary of computer buzzwords, and graphic visualizations of how microcomputers work and where they can be useful. It includes over 125 manufacturers.

Manufacturers' specs are combined with a discussion of how to use each product and how it relates to an overall computing system. Several typical systems are pictured and discussed in detail from simple home systems to large timesharing multi-user systems for business. An 8 page price list has been included for the prospective buyer.

For more information contact: W. Craig Tenney, (602) 967-1421, or write MicroAge, 1425 W. 12th Place #101, Tempe, AZ 85281.

SOFTWARE

CASSETTE MAGAZINE FOR PET COMPUTERS

CURSOR is a monthly cassette magazine of programs written just for the Commodore PET computer. Each issue contains a featured game, as well as a variety of other professionally written and tested programs for the 8K PET. CURSOR provides practical programs for business, for statistical data analysis and for use in the home. There are educational programs which children will enjoy, and some computer lore that will delight dedicated 'hackers'.

This unusual magazine is distributed each month by First Class mail on a C-30 cassette. It is available for $24 for 12 issues from CURSOR, Box 550, Goleta, CA 93017.

PET SOFTWARE

The Peninsula School Computer Project is now offering three tapes of educational programs for 8K PETs. Each tape comes with a booklet of listings and other useful information.

TAPE 1, $19.95: Pilot, Gold, Sky, Hammurabi, Names, and Hands
PILOT is an easy-to-learn programming language for computer-assisted instruction. Also included are five sample programs in PILOT.

TAPE 2, $14.95: Renumber, Lemon, Kaleidoscope, WSFN
LEMON is a simulation game in which you try to maximize profits as you run your lemonade stand. WSFN is a simple programming language for drawing figures composed of line segments.

TAPE 3, $9.95: Quest, Draw
QUEST is a difficult cave-exploration game. DRAW lets you paint pictures with the PET graphic characters.

Retail and wholesale order forms are available from Peninsula School Computer Project, Peninsula Way, Menlo Park, CA 94025. Please enclose a self-addressed stamped envelope.

WORD PROCESSOR FOR THE COMMODORE PET

CONNECTICUT microCOMPUTER announces a word processor program for the COMMODORE PET. This program permits composing and printing letters, flyers, advertisements, manuscripts, articles, etc., using the COMMODORE PET and an RS-232 printer.

Script directives include line length, left margin, centering, and skip. Edit commands allow the user to insert lines, delete lines, move lines, change strings, save onto cassette, load from cassette, move up, move down, print and type.

The Word Processor Program addresses an RS-232 printer through a CmC printer adapter. The Word Processor Program may be purchased from CONNECTICUT microCOMPUTER, 150 Pocono Road, Brookfield, CT 06804 for $29.50 postpaid.

MICROCHESS 1.5 FOR THE TRS-80

Micro-Ware Limited announces MICROCHESS 1.5, a 4K Z-80 machine language program utilizing every available byte of user RAM in the TRS-80. The program loads using the CLOAD command. Standard algebraic notation is used to describe the moves to the computer. Every move is verified for legality, and a simple command allows temporary numbering of the squares to assist in move entry.

The chess board is displayed using the TRS-80 graphics mode; pieces flash before they move, focusing attention on the move.

The program has three levels of play, challenging beginners to experienced players. MICROCHESS 1.5 is an expanded version of MICROCHESS 1.0, which has been available for 8080 and 6502 microprocessors for over a year; it incorporates many improvements suggested by users.

Other assembler and BASIC programs for the TRS-80 currently being developed include BLOCKADE, a two player game of skill and strategy, ROBOT, a computer graphics language with macro capabilities, and LIFE, the graphic cellular automata simulation. These products will be available July 1, 1978.

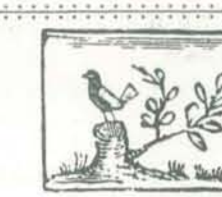
Micro-Ware products are available in computer stores or by direct mail from 27 Firstbrooke Rd, Toronto, Canada, M4E 2L2; (416) 424-1413. The price of MICROCHESS 1.5 is $19.95, postage prepaid.

TRS-80 MACHINE LANGUAGE SOFTWARE

Gary Thurmond and Tom Nussmeier have adapted a number of 8080 machine language programs and have written some new software for the TRS-80. These include the following:

- RSM-1: A 21-command machine language monitor that includes memory enter, display, search, move, exchange, zero, fill, etc., plus tape writing commands. About 2K bytes. $17.95
- RSM-1S: All of RSM-1 plus a 22nd command to display memory in Zilog symbolic code (a Z-80 disassembler)! Uses about 3.9K bytes. $23.95
- RSL-1: A graphics pattern drawing program that also plays LIFE. Patterns may be saved on tape. Takes about 2 seconds per generation. 4K bytes. $14.95
- MICROCHESS: Peter Jennings' chess playing program with graphics display. Plays a good game of chess! Fits in 4K bytes. $19.95
- ESP-1: Michael Shrayer's resident assembler, editor and monitor adapted to the TRS-80. Uses Intel 8080 mnemonics. Requires 16K byte memory.
- LST-1: A 420page disassembled listing of the TRS-80 Level I BASIC ROM's with some comments, especially for the cassette, keyboard and video driver routines. $7.95

This software runs with LEVEL I or LEVEL II BASIC, and may be ordered by mail, postpaid (California resident add 6% tax) from: Small Systems Software, P O Box 483, Newbury Park, CA 91320; (805) 494-7784 or (805) 497-6657 after 6 p m.



FOCAL FOR THE APPLE II

Apple II users can now run an extended version of the high-level language FOCAL (DEC trademark). FCL65E is available on an Apple II format cassette ($25) and occupies $1000-$2665 in memory. A Mini-Manual ($6) provides I/O information and descriptions of FCL65E's commands. The FCL65E User's Manual ($12) contains 104 pages of programming examples. The complete source listing in cross-assembly form ($35) will enable the enthusiast to customize FCL65E for his particular applications. KIM and TIM cross-assemblies, paper tapes, and KIM Hypertape cassettes are also available. For further details and information on other available 6502 software, send $1.00 to The 6502 Program Exchange, 2920 Moana, Reno, NV 89509.

SAM76 INTERPRETER

SAM76 is an interactive, high-level computer language for both home computer and time-sharing system users. It was designed to be simple enough for the novice to learn while meeting all the requirements of the most sophisticated users.

SAM76 has no error messages because there is only one data type: ASCII CHARACTERS! This means that data can be anything from complicated commands to syntax characters. SAM76 has a number of input options; for example, input may be terminated by keystroke, character count or time delay. Mathematical capability includes arbitrary precision arithmetic; it is just as easy to multiply two 100 digit numbers as it is to multiply two 1 digit numbers. SAM76 also has over 150 built-in functions that give the advanced user the capability to do such things as handle arbitrary size lists and treat them as data. A 200 page manual serves as a tutorial for the language.

Object Code is available on paper tape or TDL cassette format in the following versions:

- Z80 8K memory size, starts at 8000 hex, workspace starts at 400 hex and uses RST 0-6.
- 8080 9K memory size, starts at 2000 hex, workspace starts at 400 hex and does not use RST.
- 8080 9K memory size, starts at 2000 hex, workspace starts at 5000 hex and does not use RST.
- POLYMORPHIC version available on POLY cassette tape.
- CP/M disk version also available.

All versions are coded for APPLE/ZAPPLE conventions, but we have patch information on I/O vectors. PRICES are: $12 for SAM76 language manual, $6 for object code (except $10 for CP/M disk) and $2 for I/O information. All are available postpaid from SAM76, Box 257 RR1, Pennington NJ 08534.

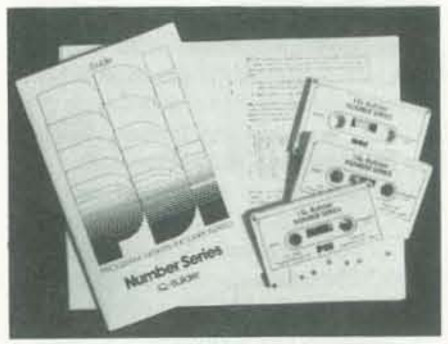
COSMAC 1802 SIMULATOR

The COSMAC 1802 simulator program enables a 6502 microprocessor to execute the COSMAC 1802 instruction set. All internal COSMAC registers are available for examination. They may be viewed statically in a single-step mode or dynamically in a trace mode. All COSMAC software features are presently supported with the exception of Direct Memory Access (DMA).

The COSMAC 1802 simulator is now available in a KIM-1 version with no additional hardware or software required. In its minimum configuration the simulator leaves two full pages of memory open for COSMAC programs. Alternative run mode features take up an additional (optional) ½ page. The simulator can be relocated in ROM and can be readily adapted to other 6502 based systems.

A complete package is available containing a KIM-1 format cassette tape, a user manual, and a complete, well commented assembly level source/object listing. Priced at $10.00 plus $1.50 postage and handling (California residents please add 6% sales tax), it may be ordered from Dann McCreary, 4758 Mansfield St, 2J, San Diego, CA 92116.

OTHER



EDUCATIONAL PROGRAMS FOR HOME COMPUTERS

Program Design Inc, a group of educational materials designers, is developing innovative educational software for the PET, TRS-80, and Apple II personal computers. Program Design courses are for preschoolers, students of all ages, and adults. Each course comes with programs on cassette tapes, printed Workbook or Guide, and any other necessary materials. Currently available courses include:

- IQ-BUILDER: Develops the skills needed to succeed on college entrance exams, civil service tests, and other aptitude tests. Each course in the series covers a specific type of exam question such as number series, analogies, and synonyms/antonyms. Programs can be purchased and used separately or together.
- STEP BY STEP: A course in the BASIC programming language that's easy even for beginners. Structured lessons and guided practice sessions are presented on the computer and in the accompanying workbook. At the end of the course, beginners will be writing their own BASIC programs.
- PRESCHOOL IQ-BUILDER: Helps 3 to 5-year-olds develop intellectual skills for learning to read.

Other courses to come include algebra, geometry, foreign languages, reading comprehension for children, speed reading. Leisure activities include menu planning, Know Your Wine, and games. For information, contact: Program Design, Inc, 11 Idar Court, Greenwich, CT 06830, (203) 661-8799.

COMPUTER COURSE FOR SMALL BUSINESS USERS

A self-instructional course, *Computer Concepts for Small Business*, has been announced by INFO 3, publishers of audio-cassette EDP courses. The course covers basic computer concepts, including types of data and how they are processed, how systems are developed, the operation of implemented systems and how to select a computer.

The course is designed to aid business-people to prepare for their first computer, by presenting the prerequisites of sound business computer applications, showing how systems are developed and operated, and covering critical management decisions like security and personnel staffing. Also, specific steps are described for evaluating and acquiring computer equipment and software.

The course contains over two hours of instructional audio-cassette tapes, plus a workbook of over 200 pages. The regular price is $140, but an introductory price of $95 will be in effect through July 15, 1978.

For more information, contact INFO 3, 21241 Ventura Blvd, Suite 193, Woodland Hills, CA 91364. Toll-free number is (800) 423-5205; in California (213) 999-5753.

GREATER GULF COAST COMPUTER CLUB

G^2C^3 has just been formed in Mobile, Alabama by a group of computer hobbyists. We get together the first Wednesday of every odd month and our next meeting is on July 5, 1978. We want to invite anyone in our Gulf Coast Area to join us at this meeting. Our present membership ranges from outright novices to established professionals, and this heterogeneousness ensures a wide variety of subjects for discussion. Members often bring their computers to these meetings for demonstrations and presentations on their building techniques. For the location of our next meeting and any additional information, please call (205) 478-1777.

COMPUTER ANIMATION FESTIVAL

For the past several years the graphics group (part of the Applied Math Division) at Argonne National Laboratory has been engaged in the development of computer graphics as a tool for use in scientific research. To help in expanding the computer animation capabilities of the Laboratory they are sponsoring the *Summer 1978 Argonne National Laboratory Scientific Computer Animation Festival*. If you would like to submit films or seek additional information please contact Eric McKinlay, Department of Computer Science, Digital Computer Laboratory, University of Illinois, Urbana, IL 61801. (Courtesy of US Computer Arts Society.)

1978 INTERNATIONAL COMPUTER MUSIC CONFERENCE

Continuing the series of computer music conferences which have been held at various places over the past few years, a conference will be held Nov. 1-5, 1978, at Northwestern University. Preliminary information is available from Peter Gena or Gary Kendall, School of Music, Northwestern University, Evanston, IL 60201, tel. (312) 492-3178.

TRS-80 PROGRAMMING CONTEST

Win $500. in TRS-80 Programming Contest. For information, send self-addressed, stamped envelope to TRS-80 P.C.-A, P.O. Box 621, Fenton, MO 63026.

CALENDAR

Amateur Computing 78, July 22-23, Arlington, Virginia. Contact Amateur Radio Research & Development Corp., PO Box 682, McLean, VA 22101.

MACC Computerfest '78, July 23-25, Detroit Plaza Hotel, Detroit, MI. Contact Michigan Computer Club, (313) 775-5320.

Personal Computing '78, August 24-27, Civic Center, Philadelphia, PA. Contact John Dilks, Rt. 1, Box 242, Mays Landing, NJ 08330. (609) 653-1188.

COMPCON 78 Fall, September 5-8, Washington DC. Theme: 'Computer Communications Networks.' Contact Harry Hayman, PO Box 639, Silver Spring, MD 20901.

WESCON 78 (IEEE et al.), September 12-14, Los Angeles, California. Contact William C. Weber, Jr. 999 North Sepulveda Blvd, El Segundo, CA 90245.

2nd National Microcomputer Exposition and Conference, September 15-17, New York Coliseum, New York City, NY. Contact Ralph Ianuzzi, H A Bruno & Associates, Inc, 78 E 56th St, New York, NY 10022.

International Microcomputer Exposition, September 29-October 1, Dallas, Texas. Contact AAME, 413 Carillon Plaza, 13601 Preston Rd, Dallas, TX 75240.

Midwest Personal Computing Exposition, October 5-8, Chicago, Illinois. Contact Midwest Personal Computing Exposition, ISCM, 222 West Adams St, Chicago, IL 60606.

Mid-America Personal Computer Show, October 12-15, O'Hare Exposition Center, Chicago, IL. Contact Austin G. Cragg, PO Box 844, Greenwich, CT 06830.

Third West Coast Computer Faire, November 3-5, Los Angeles, California. Contact Jim Warren, Box 1579, Palo Alto, CA 94302; (415) 851-7664.