

Journal of the ICCE Special Interest Group for Logo-Using Educators

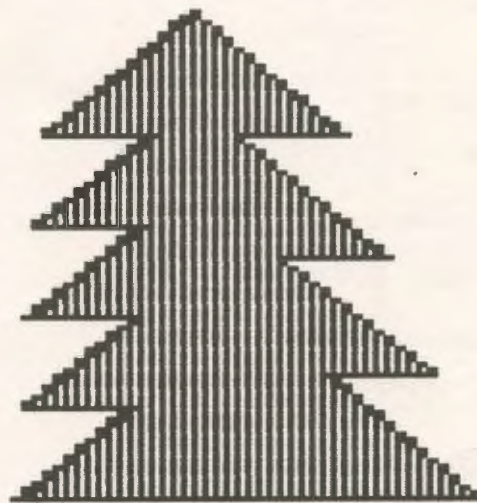


LOGO EXCHANGE

DECEMBER 1988

VOLUME 7 NUMBER 4

MERRY CHRISTMAS TO ALL



International Council for Computers in Education

✉ Long Distance Logo ✉

Educators - You don't have to go classes to earn graduate credit - let the classes come to you!

Introduction to Logo Using LogoWriter, a graduate level independent study course, allows you to learn at your own pace while corresponding with your instructor by mail.

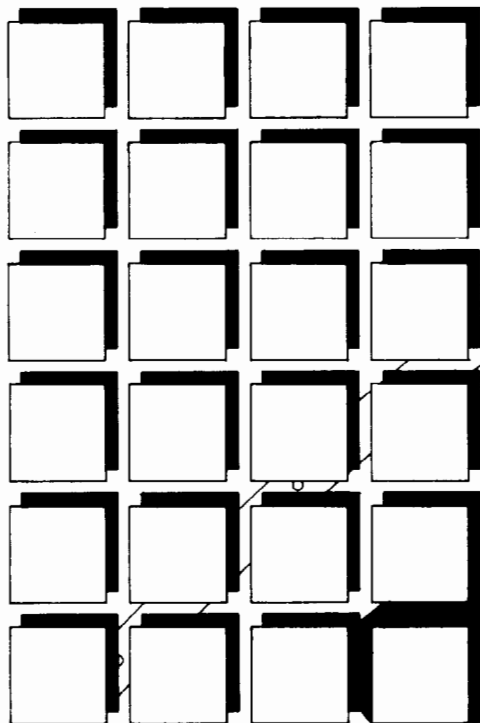
✎ **Work Individually or With a Group of Colleagues** - Take *Introduction to Logo Using LogoWriter* at home, or study with a group of colleagues at school. The course uses a combination of video tapes (ON LOGO) featuring MIT's Seymour Papert, printed materials, textbooks, and diskettes. You view the tapes, read and report on course materials, do projects, design LogoWriter lessons for your students and correspond with your instructor by mail.

☞ **Not Just Another Class** - Dr. Sharon Burrowes Yoder, editor of the *Logo Exchange* journal, designed *Introduction to Logo* to provide staff development and leadership training. The four quarter-hour course meets the standards of the College of Education at U. of Oregon, and carries graduate credit from the Oregon State System of Higher Education.

📼 **ON LOGO Video Tapes** - School Districts may acquire a license for the use of the ON LOGO package of 8 half-hour videotapes and 240 pages of supporting print for \$599.00. For a one-time fee of \$1295.00, the package may be obtained with both tape and print duplicating rights, enabling the district to build libraries at multiple sites.

➡ **Save \$\$** - If your district supports group training, *you* save money.

✎ **Tuition Information, Detailed Course Outlines, and Order Blanks** can be obtained from: LONG DISTANCE LEARNING, ICCE, U of O, 1787 Agate St., Eugene, OR 97403-9905. Phone 503/686-4414.



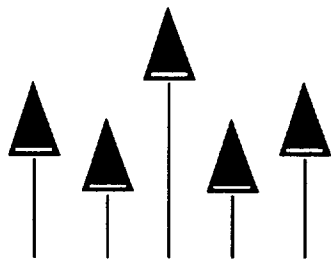
**A New
Book on Logo!**

Introduction to Programming in Logo Using LogoWriter

by
Sharon Burrowes Yoder

Designed for use either in teacher training or in an introductory secondary computer science class. Each section details new Logo primitives or programming concepts and ends with suggested open-ended activities for practice. Numerous appendices include key summaries for the Apple, IBM, and Commodore versions, a quick reference containing all LogoWriter primitives, and copies of the shapes pages. By Sharon Burrowes Yoder. \$14.95 plus \$2.50 U.S. shipping; \$3.25 Alaska, Hawaii, outside U.S. and P.O. Box shipping. All billed orders are charged \$2.50 for handling.

To order, call or write ICCE, University of Oregon, 1787 Agate St., Eugene, OR 97403-9905. Ph: 503/686-4414.



LOGO EXCHANGE

Volume 7 Number 4

Journal of the ICCE Special Interest Group for Logo-Using Educators

December 1988

Founding Editor
Tom Lough

Editor-In-Chief
Sharon Burrowes Yoder

International Editor
Dennis Harper

Senior Contributing Editor
Robb Muir

Field Editors
Anne McDougall
Richard Noss
Harry Pinxteren
Fatimata Seye Sylla
Jose Armando Valente
Hillel Weintraub

Contributing Editors
Eadie Adamson
Gina Bull
Glen Bull
Doug Clements
Sandy Dawson
Judi Harris
Gary Stager
Leslie Thyberg
Dan Watt

**International Council for
Computers in Education**
Anita Best, Managing Editor
Vincent Elizabeth Barnett, Advertising
Dave Moursund, CEO
Keith Wetzel, SIG Coordinator

SIGLogo Board of Directors
Peter Rawitsch, President
Gary Stager, Vice-President
Ted Norton, Communications
Frank Matthews, Treasurer

Publisher
International Council for
Computers in Education

Logo Exchange is the journal of the International Council for Computers in Education Special Interest Group for Logo-using Educators (SIGLogo), published monthly September through May by ICCE, University of Oregon, 1787 Agate Street, Eugene, OR 97403-9905, USA.

POSTMASTER: Send address changes to Logo Exchange, UofO, 1787 Agate St., Eugene, OR 97403. Second-class postage paid at Eugene OR. USPS #000-554.

Contents

From the Editor — <i>Hidden Gems</i> Sharon Burrowes Yoder	2
Monthly Musings — <i>Transitions</i> Tom Lough	3
What is the International Computer Problem Solving Contest (ICPSC) Don Piele	4
Little Klds and Logo -- <i>Simple Gifts</i> Leslie Thyberg	5
Logo Ideas — <i>More Conversion!</i> Eadie Adamson	7
LogoLinx — <i>Subprocedure Anti-APPENDectomies</i> Judi Harris	10
Developing Creative Writing Projects with Text Files in Terrapin Logo Mel Levin	12
Logo Connections — Software Drivers: Electronic Glue Glen Bull and Gina Bull	15
MathWorlds -- Logo and Mathematics Education (LME4) at the International Congress of Mathematics Education: A report from Joel Hillel and Cecla Hoyles Sandy Dawson, editor	18
Assessing Logo Learning in Classrooms -- III. Mathematics of Turtle Geometry -- Using 360 Degrees Dan Watt	20
Logo: Search and Research — Research on Teaching Problem Solving Douglas H. Clements	24
Global News Dennis Harper, editor	28
I Didn't Teach spelling Last Week Bill Craig	30

ICCE Membership (includes *The Computing Teacher*)

U.S.	Non-U.S.
28.50	31.50

SIGLogo Membership (includes *The Logo Exchange*)

U.S.	Non-U.S.
ICCE Member Price 24.95	29.95
Non-ICCE Member Price 29.95	34.95

Send membership dues to ICCE. Add \$2.50 for processing if payment does not accompany your dues. VISA and Mastercard accepted. Add \$18.00 for airmail shipping.

© All papers and programs are copyrighted by ICCE unless otherwise specified. Permission for republication of programs or papers must first be gained from ICCE c/o Margaret McDonald Rasmussen.

Opinions expressed in this publication are those of the authors and do not necessarily reflect or represent the official policy of ICCE.

From the Editor

Hidden Gems

by Sharon Burrowes Yoder

Have you ever made a major move across a state, country, or the world? If so, then you have some sense of the small and large difficulties that you may encounter in such situations. Of course there are the not-so-little things like where to buy groceries and how to find a good car mechanic. But there are other things that mark you as a foreigner much more quickly than lack of knowledge about the local streets and stores.

Although I have now lived in Oregon more than 6 months, there are some fine points that still mark me as an easterner (yes, Ohioans are easterners out here.) As a newcomer, there were many things to learn...how to pronounce Oregon (Ory-gun), that I live in the Willamette valley (Will-AM'-et), and that OSU refers to Oregon State (not Ohio State). But there are subtle things that take more time. For example, there's a difference between pine and fir trees (this one still eludes me) and I need to remember that the sun sets over the ocean rather than rising there (I really do get confused.) But perhaps my most noticeable failure is my inability to recognize agates. Don't laugh! This is serious if you intend to be a real ocean-loving Oregonian.

Now, this midwesterner (easterner?) thought that agates were those lovely brown colored stones that were polished and used to make things like tie tacs and earrings. Wrong. It's much more complex than that. Further, a native has to be able to recognize them when strolling along the beach. Well, I tried. "This is an agate and this isn't" said Dave showing me two sandy rocks. "Oh?" I said, not understanding. He tried again and again "Oh?" I said each time a bit more confused. After many walks on the beach over a period of months, I was still confused. I was beginning to feel that I was the brunt of an enormous joke. Agates were just rocks that the collector (Dave in particular, Oregonians in general) thought were pretty. It was a game that the natives played to make outsiders think they were doing something esoteric instead of just enjoying the ocean.

Finally, in desperation, I looked in my trusty Columbia encyclopedia. Ah, agates are a "...variety of quartz banded in two or more different colors." No problem — I knew about quartz. Hmm, that's why agates polish so well. Perhaps I had it. But no, the next walk on the beach, and I was confused again. My bewildered state was compounded by a visit to some real agate collectors. Agates all over the place, big and small and of every imaginable color. What happened to my simple banded quartz? Our hosts handed me a booklet on agates and I quickly learned that agates come in many colors and types. Why didn't my encyclopedia give me all of this information? Where was a videodisk with hypertext so that I could have explored? Well, there

really was a lot to learn. I felt better...until the next walk on the beach. I failed again — those rocks on the beach just don't look like the polished beauties in that booklet that inhabit homes and shops out here. How in heavens name do these Oregonians know that *this* is an agate and *that* isn't? To me they are simply rocks on the beach — hidden gems to my untrained eyes.

I can't help but draw an analogy between those elusive agates and the students we find in our classrooms. Yes, I know, it's pretty obvious. Some of your students are certainly mere rocks. They just sit there. Others are gems and you polish them each day with your attention. As educators we often feel that it is clear which students will benefit from our TLC and which will not. We direct our energies towards these gems — sometimes consciously, sometimes unconsciously. We let the rocks take care of themselves.

But are we really as perceptive as we believe? I'm reminded of the student who sat quietly in the back of my Intro to Computers class for the entire semester without ever saying a word. He never asked a question. He never volunteered. He never caused a problem. In fact, it took me a long time to learn his name. He was a good student, turning in all of his work on time and quite well done. It wasn't until one of the final programming projects was submitted that I discovered that Mark was a hidden gem. He produced a magnificent program that illustrated "America the Beautiful" with text, music and multiple turtle animation. I still show that project when I do workshops and presentations (perhaps you have seen it) as an example of what a high school student can do in a single semester. How many other hidden gems did I miss in my assumption that I could tell which students would polish into candidates for more advanced programming? How many of the rocks in my classroom only needed a little more TLC to shine?

Perhaps this is a good time of year to take a second look at *your* classroom. Yes, I know things get a bit crazy as the holiday season approaches. Perhaps in the craziness you'll see a special sparkle in students that doesn't show other times of the year. Take note! Perhaps you have more gems than you thought. Perhaps you have a whole classroom of gems! How about a New Year's resolution to check out your entire class this month and then polish everyone in sight come January 1st?

And now, if I could just figure out which are fir trees and which are pine trees, and which of those silly rocks are agates. I guess I'll just have to go for lots of walks on the beach, won't I?

Sharon Burrowes Yoder SIGLogo/ICCE
1787 Agate Street, Eugene, Oregon 97403
CIS: 73007,1645 BitNet: YODER@OREGON

Monthly Musings

Transitions by Tom Lough

Soon, something will happen to each of us which will change us all. We will go through a transition, the transition from 1988 to 1989. As the old year comes to a close, we will take down the calendar and its long awaited final December page, to replace it with a 12-page one for the new year. Transitions such as this provide opportunities for pausing and reflecting. I would like to take advantage of that opportunity to muse about transitions themselves.

I believe that there are important implications for teaching associated with the idea of transitions. Let me offer a few thoughts about them.

A transition involves a change from one stage or state into another. Transitions for young children can be momentous occasions. Some years ago, I was fortunate enough to observe the exact moment when my son experienced a transition which changed his life irrevocably: the occasion when he learned to crawl. Kyser, like most children, has always been the curious sort. As an infant, he was perpetually interested in what was "over there." When he learned that he could change his location by pushing up with his hands and then flopping forward, nothing near the floor in the house was safe.

One Sunday afternoon, he and I were down in the basement. While playing on the floor, he became interested in a colorful top on the other side of the room. So he flopped, flopped, flopped over to it. I happened to be observing him, and noticed that he seemed to do something extra on his last flop. Wanting to see him make that extra move once more, I placed the top on the opposite side of the room. Sure enough, he went for it. Flop, flop, flop, there! I would have sworn that he sort of crawled that last step just before getting to the top.

Being extremely curious by now, I placed the top on the opposite side of the room once more. No flopping now; he crawled swiftly over to the top and grabbed it gleefully! He had made the transition from flopping to crawling; his life (and mine!) was changed forever! He never flopped again.

Kyser's transition can be described as a permanent change affected by experience. This description contains the key elements of most generally accepted definitions of learning. But transitions have other properties which are educationally interesting.

Transitions can be one-way or two-way. By one-way, I mean that once the transition has occurred, there is no

turning back. Kyser's transition from flopping to crawling, for example, was one-way. I remember the one-way transition I experienced when Steve Tipps and Glen Bull introduced me to Logo in early 1982. I had been looking for a way to use the computer in physics teaching, but had not been encouraged by anything I had surveyed at that point. When I observed that the turtle used a self-referencing differential geometry, all my antennae went up. The clincher (I can remember it so well) was the occasion when Steve showed how to put SQUARE and TRIANGLE into a new procedure called HOUSE. Suddenly, something clicked. I could see for miles. I was never the same again, and will forever be grateful to Glen and Steve for this experience. (It was just after this experience, by the way, that the idea came to me to start a newsletter for teachers, so that information related to teaching with Logo could be exchanged and shared. If only I could figure out a good name for the publication ...)

A two-way transition permits a return to the previous state. An easy example to point out is the state of the Logo turtle's pen. You can always go from the PENDOWN state to the PENUP state and back again. Perhaps a less obvious example might be a student's transition from the concrete reasoning stage of development to the stage of formal operations. Research suggests that, under stressful conditions, students who have reached formal operations may regress to a temporary concrete reasoning state to solve particularly difficult problems. As teachers, do we experience something similar when faced with an unfamiliar or difficult situation?

Transitions can be transparent or opaque. By that, I mean that some transitions enable us to remember aspects of our former state. Like being able to see through transparent material, we can "see back" to our former state. For example, a person who has undergone the transition of marriage can remember what life was like (to some degree) before taking the nuptial vows. With other transitions, we seem to be unable to remember our former state; it is like attempting to look through an opaque object. In a classic experiment while working with Piaget, Seymour Papert demonstrated that, when children make the transition to a state in which they conserve, they are evidently unable to remember that they once said, "This container has more water than that one." When shown pre-conservation films taken earlier depicting them saying that very statement, children accused Papert of faking the footage with trick photography! The element of awareness is interesting here. I wonder what other opaque transitions we and our students undergo without being aware of them? What subtle impact might this have on our teaching and learning?

Monthly Musings -- continued

All of us have seen transitions with different speeds. Kyser's flopping to crawling transition was an example of a fast one. The "Aha! Now I've got it!" from a student is an indication of another. On the other hand, a student's complete transition from the concrete reasoning stage to the formal operations stage may take years. A student's transition from ignorance to understanding of a particular concept might take longer than a teacher expected. Are we sufficiently patient with students who are negotiating slow transitions? Are we patient with ourselves?

Sometimes, people seem to possess a sort of psychological inertia, a desire to keep things just as they are, particularly if things are generally pleasant or satisfying. In those cases, transitions can be times of stress, because of the accompanying changes. Was I sufficiently sensitive to my ninth grade science students, as they studied with me and at the same time endured the stressful transition associated with adolescence? And to the seniors in my high school physics classes, as they prepared for the transition to higher education and adulthood? I hope so. This semester, when introducing Logo to my community college physics students, I do my best to understand their concerns as they go through the transition of learning Logo and physics, as well as any other transitions they are negotiating in their personal or professional lives.

Transitions can be exciting! I am presently going through a transition which I would like to tell you about, briefly. Having served in the U.S. Army, taught high school, attained graduate degrees, and taught in a community college, I have been through some interesting and challenging transitions already. But the transition facing me this month is the most exciting of my life thus far! Effective immediately, I will be taking a position as the Curriculum Coordinator of the Educational Division of LEGO Systems, Inc. My responsibilities will include developing instructional materials for LEGO product use in science, mathematics, and other curriculum areas for grades K - 12, helping to train teachers and teacher trainers to use these materials, and supervising educational research on the results of these materials and products.

I do not intend for this transition to be opaque. The needs of real teachers in real classrooms are of great concern to me. I will continue to meet these needs in as many ways as I can, and look forward to opportunities to do so.

May each of you have a spectacular 1989, as you make the transition into the New Year. See you in January!

As ever, FD 100!

Tom Lough, LX Founding Editor
Box 394, Simsbury, CT 06070

ICPSC

What is the International Computer Problem Solving Contest (ICPSC)?

To some schools it is a chance to challenge their top computer and mathematics students in an annual computer problem solving event that compares their solutions to the best in the world. To others it is a challenging set of problems to be used as enrichment material for computer programming classes. However you choose to use the ICPSC, it can be a valuable resource for any programming teacher.

What is unique about the ICPSC?

The ICPSC combines the art of problem solving with the skill of computer programming. Our contest challenges teams (from one to three students each) to create short, original solutions to a set of five problems within a two-hour period. The problems fall into five categories: (1) computation, (2) simulation, (3) graphic patterns, (4) words, and (5) mind benders.

Can an individual teacher in a school become a contest director for his/her school?

Yes! We want to challenge and enrich your students so you don't have to make it a major district-wide event if that would stop you from doing it. Of course the more students in your local area who can get involved the better. But, if you only have one student in your whole school who wants to compete, that is all you need.

Can Logo be used?

Our original contest problems were not appropriate for Logo students. So we added a new Logo contest and last year expanded this contest to include all three age groups.

What are all the age groups?

Elementary Division (Grades 4-6), Junior Division (Grades 7-9) and Senior Division (Grades 10-12) are now offered in either the Open contest (all languages) or the Logo contest.

When is the contest held?

The 9th Annual ICPSC is scheduled for Saturday, April 29, 1989. Friday, April 28 and Monday, May 1, are alternate dates that can be used if Saturday is impossible. This year's set of contest problems will be mailed to the contest director on April 1, 1989. Sets of previous problems and solutions for students to practice on are always available.

How can I get more information about the ICPSC?

The Computing Teacher annually publishes articles about the contest in the October, November, December issues. Also the *Logo Exchange* is now publishing stories on the Logo contest. You may receive a free copy of *Compute It!*, the official newsletter of the ICPSC which is published in November each year, by writing to:

Dr. Donald T. Piele, ICPSC
P.O. Box 085664, Racine, WI 53408
Ph 414-634-0868

Little Kids and Logo

Simple Gifts by Leslie Thyberg

Deck the walls with a children's keyboard
 Fa-la-la-la-la-la-la-la-la
 Tis' the season for learning FORWARD
 Fa-la-la-la-la-la-la-la-la
 Don' we now our Turtle apparel
 Fa-la-la-la-la-la-la-la-la
 As we sing of learning pitfalls and perils
 Fa-la-la-la-la-la-la-la-la

Logo Lesson

A Logo learning environment is no different than any truly stimulating classroom. Simply put, it is a place that allows for children to explore, discover, and build their own intellectual structures with materials drawn from their surroundings. So too, a Logo lesson runs much the same as any other lesson plan might.

1. A new concept is introduced.
2. When appropriate, review and reteaching are done to tie the concept in with what is already known.
3. Examples are practiced together as a group.
4. Guided practice occurs.
5. Free exploration takes place.

All lessons should proceed from the familiar to the unfamiliar; the known to the unknown; the concrete to the abstract. The individual strengths, needs and differences of each child should always be kept in mind. For example, perhaps you are working on some keyboard familiarity with your students. The following is a sample Logo lesson using this format:

The new concept: Orientation to the keyboard

Specific objectives:

1. The students will be able to locate the following keys on the keyboard: F, D, B, K, L, T, R, all numbers, Return/Enter, Esc (or Delete), and the Space bar.
2. The students will be able to differentiate between the letter O and the number 0, and between the letter L and the number 1.

Group practice:

To orient the children to the keyboard, have a discussion about its similarity to a typewriter. Use an actual typewriter for demonstration and comparison. Introduce the keyboard using a poster-size replica or an overhead projection. Show all the letters and the numbers. If you're more secure as a ditto user, pass out dittoed copies of the keyboard to all the students. Then, as you have various children come to the poster or overhead display and press the "keys" to carry out specified commands, all the students can role play this in

their seats with their individual keyboard replicas. Then, gather the group around a computer. Select one volunteer at a time to sit at the computer and have the group dictate which key to use, until many of the children have experimented with a variety of keys.

Guided practice:

Remember the game Twister? You can use an actual Twister mat, a checkered plastic table cloth, or you can label squares on the linoleum floor to get your students keyed in and kinesthetically involved. Too chaotic or crowded for your teaching style? I employed the zealous energies of some older students to make me some small Bingo-like game boards for a game we called "ERTLE." On the boards, we placed abbreviations for the most frequently used primitives. Of course, just like Bingo, ERTLE has HOME as its center square. It was a highly motivational, very inexpensive way of giving the children lots of exposure and review so that when their altogether too limited computer time rolled around they could utilize it to the fullest.

Prefer individual or paired peer practice? Patricia Mocello, a second grade teacher in the Pittsburgh Public School system, cut out tagboard ladybugs. She called the game "BUG and DEBUG." She made 18 cards with Ladybugs on one side and computer bugs and debugs (e.g., missing spaces, misspellings, etc.) on the other side. The objective was to have the student match the bug and the debug. Much like concentration, the students mix up cards and spread them out face down. They are to turn over two cards and try to match the bug with its debug. If they match, they take the pair. If they don't match, turn them back over.

Several exemplary teachers from past workshops that I have conducted put together some wonderful learning centers for their students. One made a huge, felt monitor screen with the grid lines filled in. She cut out colorful arrows and turtles for her young students to freely play with and explore. There were also specific task cards with open ended exploratory activities for the students to engage in as well. Another creative teacher made a velcro-keyboard with a similar set of activity cards. Being budget minded and totally inept with a sewing machine, I chose to use the burger containers (e.g., for a Big Mac) and invert them on the wall as a combination bulletin board display, visual aid, and learning center. The carton bottoms were covered with construction paper squares and labelled according to the kind of machines in the classroom.

Other Off Computer Activities

Most classrooms do not have the luxury of one computer for every student. More often than not, the situation is one computer and thirty students or an infrequent visit to the

Little Kids and Logo -- continued

computer lab. To maximize hands on time, the use of off computer or precomputer activities is strongly suggested. These activities help give the students direction and purpose.

Myrtle says:

No Logo classroom is complete without a turtle mascot. Furry Folk ® make wonderful puppets. Their turtle does a great HIDE-TURTLE retreat into its shell. Trying to help your students differentiate between left and right? Use a turtle mascot (puppet, stuffed toy, or engineer's cap on your head with turtle appendages attached!) Introduce Myrtle the mascot and tell the children that the turtle will be giving them directions in the Myrtle Says Game. Play the game just like Simon Says. Give each child a set of LT and RT stickers or tie colored yarn around their right wrists. Show the children on the board or monitor screen the abbreviations for LEFT and RIGHT and FORWARD and BACK. Play Myrtle Says giving oral directions or by using flashcards. Let several children take turns giving the directions.

Playing Turtle

The advantage of your taking the role of the turtle is that you can model error messages for your students. Take the game a step further and include your keyboard center. Have students take turns typing the commands on the keyboard. Have them verbally relate what they are typing "Forward Space bar Fifty", "Right turn Ninety Space bar." and so forth. This way, if they neglect to leave a space, you can model with the response I DON'T KNOW HOW TO... It helps them learn to debug their own problems, and to view errors as helpful and playful prompts from the turtle.

Treasure Hunts

Since repetition and experience are two trademarks of early learning, another way to help your students learn to give and/or receive commands is to send them on a Turtle Treasure Hunt. Much like Button Button, Who's Got the Button, you can choose a child to leave the room while an object is being hidden. Then select volunteers from the group to give commands to the child when the child comes back into the room. Let them give commands until the object is found. After each round, discuss the efficiency and correctness of commands given and carried out.

Take the treasure hunt a step further. Have a 5 minute hunting spree for all the square (round, rectangular, triangular, etc.) shapes that they can find in the classroom. (For homework they can go on a shape hunt. It's amazing how many bathroom floors suddenly become objects of interest when Logo is being learned!) Have a shape of the day (or of the week — whatever time frame your students require.) In the art center have lots of construction paper cutouts of that

shape. If its squares, have them create a picture using different sizes and colors of squares. The ultimate challenge comes when they have their computer time. Can they make that same picture by drawing it with the turtle? Maybe you've covered all the shapes by now. Have the students create geometric stick figures. In the writing center have them write stories (or take dictation from them) about their creations. Can they make a Logo stick person using the shapes they used in the art center? This is a great springboard into subprocedures. But I'm getting ahead of myself. My On Computer Gifts and Utilities are coming your way in a future column!

Dr. Leslie F. Thyberg
5637 Rippey Street
Pittsburgh, PA 15206
or
c/o Chatham College
Woodland Road
Pittsburgh, PA 15208
AppleLink ALS 038

The ECCO Logo Project: *Materials for Classroom Teachers and Teacher Trainers*



*Edited by
Theodore C. Burrowes &
Sharon K. Burrowes*

This book presents Logo activities for use in grades four through eight. Use it for teacher training or take it directly into the classroom to enhance the teaching of language arts, social studies, science and math.

The ECCO Logo Project Includes: Student worksheets, teacher-training materials, Logo II version of all materials, and Apple Logo version of teacher materials.

The Educational Computer Consortium of Ohio (ECCO) developed these materials over the course of an academic year, in conjunction with its extensive series of Logo workshops.

To Order, write or call:
ICCE, University of Oregon, 1787 Agate St, Eugene, OR
97403-9905, or phone 503/686-4414.

An International Council for Computers in Education Publication

Logo Ideas

More Conversions!

by Eadie Adamson

I was so delighted to read Judi Harris' September LX column on "Secular Conversions" just as I was procrastinating over finishing this column (yes, we all really do write that far ahead!). This month's "Logo Idea" is about yet another conversion or, better yet, a *link* between versions of Logo and the pictures that they produce.

Increasingly you will find that newer programs create files that can be easily accessed by other programs. This has become very important to me. I write with AppleWriter, save LogoWriter 2.0 programs as text and import them into my AppleWriter document, and then occasionally load the assembled file into MultiScribe on my GS and "prettify" it before printing. I also send ASCII files created by AppleWriter to Sharon Yoder which she transfers into Microsoft Word on the Macintosh for editing. She then loads the Word files into PageMaker to so she can typeset them for the *Logo Exchange!*

While this interchange of files has been possible with LogoWriter text files and other word processors, as well as with LogoWriter and some graphic programs, a similar connection between LogoWriter and Terrapin Logo pictures has not been so easily achieved until now.

Now Terrapin has a new ProDOS Logo called Logo PLUS, which I first played with at the Logo Institute this summer. What interested me about Logo PLUS was that a picture saved in LogoWriter can be loaded into Logo PLUS, changed, then saved again (this time in Logo PLUS), and finally be loaded back into LogoWriter to be animated with multiple turtles. Why, you might ask, would you want to do this? Read on....

New Features in Logo PLUS

Logo PLUS has a variety of text styles, something not available in LogoWriter. The text is displayed as graphics, comparable to LogoWriter's LABEL mode. Since the operating systems of Logo PLUS and LogoWriter are the same (Apple ProDOS), it is possible to access special features of either version by saving a picture and then loading it into the other version of Logo for additions. For a LogoWriter user, this means that the various text styles of Logo PLUS — boldface and italic, a different style of plain text, as well as options to underline or use color — are now available for labeling pictures, creating signs and the like. The final version of Logo PLUS has a font editor as well. Think of the wonderful maps and city scenes you can make now!

The Switch

The interchange process between LogoWriter and Logo PLUS is much easier than what one must do to import or export between Logo and Print Shop (see Judi Harris, *September 1988 LX*, pp. 15 - 17) — and every bit as much fun!

Here's all you need to do to get a LogoWriter picture into Logo PLUS:

With LogoWriter loaded and your picture on the screen, type:

```
SAVEPIC "name.PICT
```

The .PICT is essential for Logo PLUS, which identifies picture files by the .PICT at the end of the file name. Without .PICT Logo PLUS cannot identify your saved picture as a picture file.

Press RETURN.

Your picture is now saved as a binary file on your disk.

Load Logo PLUS.

Type:

```
READPICT "name.of.picture
```

Omit the .PICT here; it serves only as an identifier for pictures in Logo PLUS.

Press RETURN.

Behold! A LogoWriter picture on a Logo PLUS screen. Now you can add all the fancy text you want using Logo PLUS's special commands for adding text.

When you finish adding the features you need from Logo PLUS, it's time to save the picture again:

```
SAVEPICT "name.of.picture
```

You don't add .PICT this time; Logo PLUS does this automatically. Unlike LogoWriter you can use the same picture name in Logo PLUS. Logo PLUS will overwrite the old picture with the new version.

Now reload LogoWriter. Type:

```
LOADPIC "picture.PICT
```

This time you need the .PICT since LogoWriter takes its file names literally. If you catalog the disk on which you saved the picture — use SHOW or PRINT FILELIST if you are in LogoWriter — you will see the picture listed with its name and .PICT appended. By the way, you can save your pictures

Logo Ideas -- continued

on any disk formatted with ProDOS. You don't have to save your pictures on a LogoWriter or Logo PLUS disk!

Go ahead and use your new picture!

Notice that Terrapin READs and LogoWriter LOADs a picture. Just a little confusing, but you can manage. Terrapin also uses PICT and LogoWriter uses PIC: SAVEPICT or READPICT for Logo PLUS, LOADPIC or SAVEPIC for LogoWriter. If you associate "T" with Terrapin you'll get it right every time.

Two Printing Sizes in Logo PLUS

Another interesting possibility is presented by Logo PLUS's two screen printing sizes: a standard picture size and a reduced small screen version. When I tried Logo PLUS I thought immediately of how useful this interchange could be to a confirmed LogoWriter user (yes, I still am — but that's another issue.) Think of the illustrated stories students have written. Now they can be printed in small size, text can be printed as well, and neat little books can be put together. True, you could also reduce your pictures on a xerox machine, but this is much more fun!

A Different Kind of Shape and Screen Editor

The shape editor for Logo PLUS allows you to make any size shape you want, even full screen size. Rather than filling in a rectangle with squares, Logo PLUS works more like Pixmap than anything else I can think of. (Macintosh users would talk about editing "fat bits.") Much more finely detailed shapes can be created than with LogoWriter. You see the shape you are designing, as you do in the new IBM Logo-Writer.

While there is still only a single turtle, the shapes you design in Logo PLUS will rotate unless you use the LOCK-HEADING primitive which prevents the shape from turning. Using the VFLIP and HFLIP primitives, the entire screen can be flipped in the immediate mode or under program control (in procedures.) Shapes can also be flipped horizontally or vertically in the shape editor, making it easy to copy and reverse a shape at the same time.

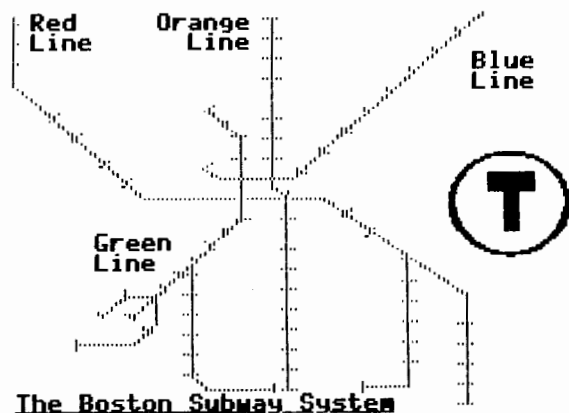
Shapes can be stamped with two different commands: STAMP and STAMPXY, which takes x- and y-coordinates as inputs. The Logo PLUS shape editor is not fully developed in the pre-release version I have been playing with, but it is to be a screen editor as well. Imagine saving a student's work and then giving him the capability of fine tuning a messed-up picture. In addition you can take whatever is already on the screen and turn that into a shape as well.

If a Terrapin user wants to use stamped LogoWriter shapes in a picture, it is perfectly possible to do so by saving the Logo PLUS picture, loading it into LogoWriter, creating

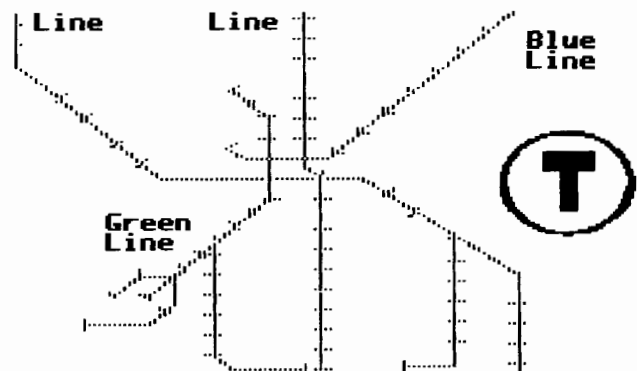
and stamping appropriate shapes, and then saving the picture (with a new name) to reload into Logo PLUS. A LogoWriter picture can also acquire added shapes from Logo PLUS if you can be satisfied with stamped shapes. Unfortunately, the two shape systems are not compatible. You cannot create shapes in Logo PLUS and animate them in LogoWriter. The reverse is possible, however. Save a page with LogoWriter stamped shapes, read the page into Logo PLUS, use special Logo PLUS editing commands that allow you to edit the whole screen and turn a part of the LogoWriter stamped shape into a Logo PLUS shape, and there you are!

Some Examples

As I thought about all these possibilities, I decided to try some things for myself. Here, for example, is a screen print of a Logo Plus map of the Boston T (Boston's subway system), which Dorothy Fitch of Terrapin, Inc. created as a starter for a group project at the Logo Institute this summer. I have added text to show the bottom of the Logo PLUS screen:



Here is the same map after loading into LogoWriter:



Notice that the top and bottom lines of the text are lost in the transfer to LogoWriter. That would need to be taken into account when working back and forth between versions.

Here is a sample of a LogoWriter page which was saved and then loaded into Logo PLUS. The italicized lines show you what can be added, in Logo PLUS, to a full-screen LogoWriter picture.

This line is added from Logo PLUS!!
 ➤ An animated plane flight
 with three turtles.
 TAYLOR GRANT
 Grade 4 ➤



And this line is added from Logo PLUS!!

And here is that same picture saved from Logo PLUS and loaded into LogoWriter. It looks just as it did originally; the added Logo PLUS lines do not show.

➤ An animated plane flight
 with three turtles.
 TAYLOR GRANT
 Grade 4 ➤



THIS IS HOW THE PICTURE SAVED
 IN LOGO PLUS LOOKS WHEN LOADED
 INTO LOGOWRITER. THE TOP AND
 BOTTOM LINES OF TEXT ARE GONE!

Think of the Possibilities!

I have a number of students whom I know will be fascinated by the possibility of interaction between LogoWriter and Logo Plus. Here are some potential projects which might interest them — or you:

A STORYBOOK

Make a picture story in LogoWriter, design a beginning reader or an alphabet book — use graphics only.

Save the pictures.

Load Logo PLUS and, one at a time, load the LogoWriter pictures and work on them.

Use Logo PLUS to add simple text: a single line at top and bottom if your LogoWriter picture has filled the entire screen. (You could leave additional space at top or bottom for addition of more text from Logo PLUS.)

Print the pictures in the smaller size to make a book.

AN ANIMATED STREET SCENE OR MAP

Take your favorite LogoWriter street scene or map.

Save the picture.

Load the picture into Logo PLUS.

Add fancy text labels to your picture.

Save the picture.

Load the picture into LogoWriter and add your animation.

(Or, if you are a Logo PLUS partisan, make wonderful new shapes with Logo PLUS and animate one of them.)

HIGHWAY SIGNS

Design a billboard.

Create shapes to stamp on the billboard.

Decide on your text.

Use the variety of text styles available in Logo PLUS and the

LogoWriter text to create an interesting sign.

Make the LogoWriter text flash!

A MAP-MAKING PROJECT

Start a class map-making project.

Design shapes to indicate topography.

Leave some labeling of the maps and perhaps some of the topography symbols for Logo PLUS.

Save the LogoWriter picture.

Load into Logo PLUS.

Add the rest of the text for the map, in appropriate colors and type styles.

Save the picture again.

Load back into LogoWriter.

Add some flashing signs at key points of interest.

Program an animated "tour" of your map:

An interactive program might allow you to "drive" through the map, a nice project for both LogoWriter and Logo PLUS users.

An automated tour might give you tour guide information in the LogoWriter command center. Use TYPE to place text in the command center. Use TYPE CHAR 13 to add a carriage return at the end of the text.

So, if you're a LogoWriter user, it will be great fun to have a few copies of Logo PLUS around. If you are a Terrapin user, switch to Logo PLUS and get a few copies of LogoWriter to try this whole process in reverse!

Now, if only we could have different text sizes too ... or multi-colored shapes

... or a version of Logo that make use of the sound and color capabilities of the GS or

Is anyone out there listening?

Note of thanks: I want to thank Dorothy Fitch of Terrapin, Inc. for allowing me to use a test version of Logo PLUS to write this article. Her help on technical questions was invaluable.

Eadie Adamson, Allen Stevenson School
 132 East 78th Street, New York, New York 10021

Logo LinX

Subprocedure Anti-APPENDectomies by Judi Harris

I once heard of a chemistry professor that immediately erased with his left hand what he wrote on the chalkboard with his right. The idea fascinated me, although I was thankful that I wasn't in his class. Logo work in the immediate mode is a bit like the professor's quick-to-disappear chemical equations. The result may be admired for its precision, beauty, or conceptual power, but the process by which it was formed has disappeared into chalkdust or computer ether.

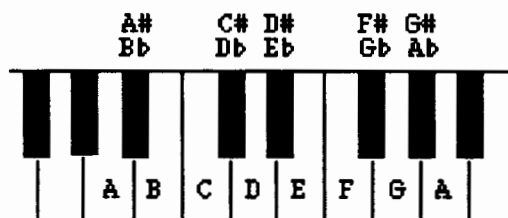
Logo novices may be hesitant to do what you are about to suggest: write procedures that can be stored in memory and invoked with a single command. The leap from immediate mode doodling to preplanned, hierarchically structured procedure writing can be a large one.

It is possible to bridge these two experiences with a simple Logo tool that I call APPEND. When added to all student- or teacher-defined procedures, APPEND constructs a sequential record of subprocedures as they are used in the immediate mode. This list can then be defined as an operable superprocedure with another tool procedure called TITLE.

A Melodious Microworld

Consider, for example, a project that a Pennsylvania fifth-grader conceived and coded. The idea of Logo music intrigued him, especially because he was taking clarinet lessons, and reasoned that if he wrote Logo procedures to play practice pieces for him, then he would be sure to play the pieces correctly on the clarinet.

He proceeded to make eighteen subprocedures, one for each note in the chromatic scale beginning with A below middle C. All shared a common local variable, TIME, used to express the duration of the notes.



```

TO A :TIME
TONE 440 :TIME
END

TO B :TIME
TONE 494 :TIME
END

TO C :TIME
TONE 523 :TIME
END

TO D :TIME
TONE 587 :TIME
END

TO E :TIME
TONE 659 :TIME
END

TO F :TIME
TONE 698 :TIME
END

TO A# :TIME
TONE 466 :TIME
END

TO Bb :TIME
TONE 466 :TIME
END

TO C# :TIME
TONE 554 :TIME
END

TO Db :TIME
TONE 554 :TIME
END

TO D# :TIME
TONE 622 :TIME
END

TO Eb :TIME
TONE 622 :TIME
END

TO F# :TIME
TONE 740 :TIME
END

TO Gb :TIME
TONE 740 :TIME
END

TO G# :TIME
TONE 830 :TIME
END

TO Ab :TIME
TONE 830 :TIME
END

```

(Note: LCSII Logo II and Apple Logo II use the command TOOT instead of TONE, but the frequency numbers are the same.)

The student could then easily combine the subprocedures into superprocedures that played practice songs.

```

TO CHROMATIC.SCALE
A 24
A# 24
B 24
C 24
C# 24
D 24
D# 24

```

```

E 24
F 24
F# 24
G 24
G# 24
A2 24
END

TO TWINKLE
A 12 A 12 G 12 G 12 A2 12 A2 12 G 24
F 12 F 12 E 12 E 12 D 12 D 12 C 24
G 12 G 12 F 12 F 12 E 12 E 12 D 24
G 12 G 12 F 12 F 12 E 12 E 12 D 24
A 12 A 12 G 12 G 12 A2 12 A2 12 G 24
F 12 F 12 E 12 E 12 D 12 D 12 C 24
END

```

But when the student's younger brother stopped by after school, it was clear that he was not ready to write tune procedures like the one above. Instead, he began playing with the note subprocedures in an immediate mode format. His frustration was, of course, that a particular sequence had to be recorded on paper or memorized to be repeated.

Subprocedure APPENDages

An APPEND tool at the end of each of the note procedures would have automatically built a sequenced list of the notes that the younger brother used in his melody experiments.

```

TO APPEND :COMMAND
MAKE "CREATION LPUT
  :COMMAND :CREATION
END

TO A :TIME          TO A# :TIME
TONE 440 :TIME      TONE 466 :TIME
APPEND "A           APPEND "A#
APPEND :TIME        APPEND :TIME
END                 END

TO Bb :TIME
TONE 466 :TIME
APPEND "Bb
APPEND :TIME
END

```

In the APPEND tool, CREATION is a global variable name that refers to the record of commands, in list format, that forms sequentially in the computer's memory as the music program user builds a melody. CREATION must be set to [] (an empty list) before the musical record of user keystrokes can be started.

```
MAKE "CREATION [ ]
```

Another simple tool procedure, TITLE, takes the current value of CREATION and defines it as a procedure with a title that the user chooses. It then restarts "CREATION, so that the next tone sequence may be recorded separately.

```

TO TITLE
CLEARTEXT
TYPE [TITLE?]
DEFINE READWORD LIST [ ] :CREATION
MAKE "CREATION [ ]
CLEARTEXT
END

```

You're probably thinking, "but what happens if the user types a note, then wants to change it?" As a devoted facilitator, I'd reply, "that's a fascinating programming challenge! How might you go about coding a tool to allow users to alter the string of commands in the computer's memory?"

INSTANT History

One of the most popular examples of action-recording microworlds is INSTANT, a single-keystroke adaptation of Logo graphics commands for new or physically challenged users. Most versions of INSTANT allow users to enter turtle commands with single letters (i.e., F(orward), B(ack), R(ight), and L(ef)). This limits the number of commands that can be created to the number of characters on the keyboard, and further constricts inputs to these commands. Most single-keystroke INSTANTs, for example, make the turtle move 10 steps forward when the user presses an F, and turn the turtle 10 or 90 degrees to the right or left when users type R or L. Angle estimation and length approximation become much less conscious activities when users don't have to type numbers as inputs to movement or heading commands.

If users are physically able to press the Enter or Return key after typing a command, then command options can be greatly increased. Microworld function names can be one or more letters long, can accept variable input, and if APPEND is used, an automatic record of user actions can still be maintained. Moreover, new commands (in the form of short procedures) can be added to the microworlds by either students or teachers without having to change existing code.

APPENDix of Ideas

Although the appropriateness of using INSTANT-like microworlds has been debated (see the articles listed in

Logo LinX -- continued

the References at the end of this column), user-based, self-recording microworlds need not be limited to INSTANT-like programs. Microworlds are best used when they are integrated into subject-centered study. Here is a list of possibilities. Hopefully, these ideas will spark others that can be infused into curricular study in your classroom.

- rebus maker (language arts)
- hieroglyphic typewriter (social studies)
- chemical element diagram-maker (science)
- pattern block microworld (mathematics)
- animation tool (SETSHAPE sequences; art)
- Lego Logo obstacle course path for a car (science)
- sound effect sequences (music)
- braille, Morse code, or sign language translator (language arts)
- choreography tool (music/dance)

If you would like to see examples of how procedures like APPEND and TITLE can be used in subject-centered microworld construction, I would be glad to send you a copy of a pattern blocks/parquetry blocks microworld that I constructed for elementary school use and/or an upgraded version of INSTANT, written for kindergarten students, that I call DRAW. Send a blank, double-sided diskette and a disk mailer with sufficient return postage to the address listed at the end of this article. I would also love to hear from you if you create or revise microworlds using procedures like APPEND and TITLE.

References

- Bull, G. L. & Cochran, P. S. (1986). Instant tools. *National Logo Exchange*, 4(4), 8-11.
- Lough, T. M. (1983). Slow turtle moves clearly. *National Logo Exchange*, 2(1), 7-8.
- Tipps, S. (1983). The issue of instant. *National Logo Exchange*, 2(4), 6-10.
- Watt, D. & Watt, M. (1986). Starting Logo with young children? Slow down the turtle! *Logo Exchange*, 5(1), 3-5.

Judi Harris taught students in Philadelphia-area elementary through graduate schools to use computer in teaching and learning for six years. She now does similar work at the University of Virginia, where she is completing her doctoral work in Instructional Technology. She can be reached at

Judi Harris
621F Madison Avenue
Charlottesville, VA 22903
CIS: 75116,1207 BitNet: jbh7c@Virginia

Creative Writing

Developing Creative Writing Projects with Text Files in Terrapin Logo

by
Mel Levin

Using Logo as a Text Editor

The Logo language is designed to read and save files that contain Logo procedures. By modifying the system, one can use Logo to read and save text files (and thus be able to use the Logo editor purely as a text editor).

The advantage to using the Logo editor as a text editor rather than as a procedure editor is two-fold. First, writing text in this manner uses very little memory in the workspace. This is helpful if you are writing a program which contains a lot of text material (instructions, explanations, written passages, etc.) If you wrote the text as procedures, you would be using primitives which take up lots of memory in the Logo workspace. If your program is large, there is danger that you might encounter the dreaded "no storage left" message. Using the Logo editor as a text editor helps eliminate the problem.

Secondly, you are using the Logo editor as a quasi-word processor, which means that you write text as you would in English and not as Logo code. This means there is no code syntax problem (no hassles with PRINT, SENTENCE, WORD, [,], etc.). You also have the freedom to format the text as you would with a word processor. This is not to imply that the Logo editor has terrific word processing capabilities. It is, in fact, quite limited as a word processor. The point, however, is that text can be formatted relatively easily within the restrictions of the Logo environment.

Normally, to enter the editor you type TO followed by the name of a procedure to be created or edited. To work with just text in Logo, it is necessary for the edit buffer in memory to be empty. Entering the edit buffer is achieved by typing TO followed by Return. At this point text is written and edited the same way as a procedure is written and edited. You use the same cursor keys to move around the editor. In order to save the text, it is necessary to exit the editor. Instead of typing Control-C, which defines a procedure, Control-G is used. This exits the editor without modifying what is in the edit buffer. Thus, the text will be printed as written, and not reformatted as would a procedure. Next, you must save the buffer as a file on the disk. At this point, you must not edit any procedures or enter the graphics mode before you save the buffer or it will be lost. While you can save the text with any name, you cannot save in the usual way.

Fortunately, there exists a group of procedures that allow you to save text easily. The procedures, called TEXTEDIT, from Terrapin Logo are on the Utilities I disk that comes with the Logo language.

Here's how to use the procedures to write and save text files:

1. First read the TEXTEDIT files into computer memory.
2. Type TO followed by Return. This allows you to use the Logo editor as a text editor.
3. Type text as you would in a word processor.
4. Exit the editor with Control-G.
5. Type SAVETEXT "FILENAME immediately.

Armed with the TEXTEDIT procedures and the knowledge of how to use them, let's explore the concept of developing a creative writing project.

Writing An Interactive Text Adventure

Logo can be incorporated across many curriculum areas. In the area of Language Arts, for example, Logo can provide an interesting forum for developing creative writing. One way the Logo editor and the accompanying procedures can be utilized is to write interactive text stories. Students can write the stories in Language Arts, and then use Logo code to produce interactive text stories on disk.

The concept of interactive creative writing is to permit the reader (user) to be an active participant in the story. As s/he reads, s/he makes decisions that affect the story line.

The first step in this project is for students to prepare their stories off the computer. I have had teams of students write the story together as a group. They decide on the story plot and write each of the sections. It is important that students follow a prescribed writing process. The Language Arts teacher can be a collaborative support person here providing the various techniques and strategies. We then put the pieces together to complete the story, making sure each branch goes where it should (use tree diagrams, etc. to graph the story board).

Plan each screen of text as a separate page in a book. Give each page a name, title, or number to identify it. At the end of each page, direct the user to make a decision. This will send the user to a different page of the story, thus creating many different story plots.

When you have students develop their interactive writing, it is important that you have them chart the course of the story. This can be achieved by introducing the concept of

tree diagrams, flow charts, etc.. It will make things easier when you begin to code and certainly aid in the writing itself.

After the story flows correctly, we then begin to code it in Logo and save it on disk. Theoretically, you can develop a complete library of stories on disk for everyone to share and enjoy!

Lastly, we need to write a number of procedures to act as a template, in order to control the correct flow of the text files. Unless the students have a great deal of experience in Logo, they will not be able to write them themselves. However, you can write the template for them.

Sample Interactive Story

This creative writing project was written by a team of four sixth grade students a Prince Hall School they called their story *The Attic Adventure*. (See the procedure tree on the next page.)

Template Used

Here is an example of a template of procedures that I used for the creative writing project.

```

TO START
MAKE "STORYEND? "FALSE ;—>
  Sets a flag that will be
  checked in subprocedure SELECT
CLEARTEXT
MAKE "PREFACE [INTOR]
MAKE "INTRO [STAYOUT CASTLE]
MAKE "CASTLE [NO.APPLES APPLES]
MAKE "APPLES [WITCH NO]
MAKE "NO.APPLES [EAT NOT.EAT]
MAKE "WITCH [ESCAPE DELAY]
MAKE "DELAY [RED.DOOR BLUE.DOOR]
MAKE "BLUE.DOOR [CHEST SHUT]
MAKE "FILE "PREFACE ;—> Makes
  the variable name "FILE have the
  value "PREFACE. This is the name
  of the beginning page.
SHOWFILE "PREFACE ;—> Prints
  the file on the screen
SELECT ;—————> Subprocedure
  that asks for the user to respond
  with a decision.
END

TO SELECT
MAKE "CHOICE READWORD
TEST :CHOICE = "QUIT
IFTRUE CLEARTEXT PRINT [HOPE YOU
  PLAY AGAIN!] STOP

```

Developing Creative Writing Projects with Text Files in Terrapin Logo -- continued

```

TEST MEMBER? :CHOICE THING :FILE
IFFALSE SELECT.AGAIN
CLEARTEXT
IF MEMBER? :CHOICE STORY.ENDER
    MAKE "STOREND? "TRUE
SHOWFILE :CHOICE
IF :STORYEND? = "TRUE STOP
MAKE "FILE :CHOICE
SELECT
END

TO SELECT.AGAIN
PRINT [WHAT DO YOU WANT TO DO?]
PRINT SENTENCE THING :FILE
    [OR QUIT?]
MAKE "CHOICE READWORD
IF :CHOICE = "QUIT CLEARTEXT PRINT
    [HOPE YOU PLAY AGAIN!] TOPLEVEL
IF MEMBER? :CHOICE THING :FILE STOP
SELECT.AGAIN
END

TO STORY.ENDER
OTUPUT [EAT ESCAPE CHEST NOT.EAT
STAYOUT RED.DOOR NO SHUT]
END

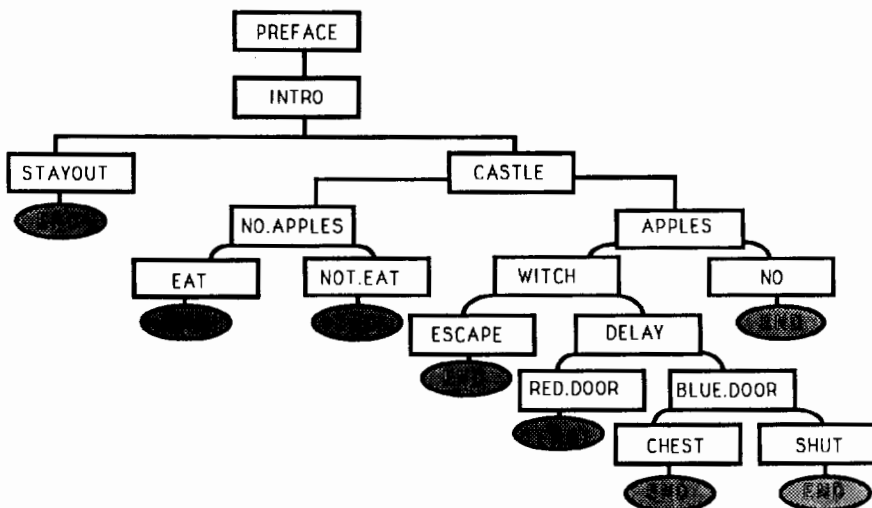
TO READWORD
MAKE "CHOICE REQUEST
IF :CHOICE = [] OUTPUT READWORD
OUTPUT FIRST :CHOICE
END
    
```

A Couple of Final Thoughts

- You can stop the flow of text across the screen with Control-W to aid any slow readers. Pressing any key will resume the flow of text.
- The students don't really have to understand how the procedures in the template work to enjoy and get involved in writing creative stories and then using Logo code to develop an electronic library. The tool procedures can be viewed as mysterious black boxes that are to be enjoyed because they do the job (i.e., radios, TV's, automobiles, etc.) without worrying how they operate.
- Students can add Logo graphics to enhance or embellish the story by plugging a procedure or two into the SELECT procedure. (I leave that problem for you to play with.)
- The idea of having their work electronically preserved excited and delighted the students involved.
- The final product was read over and over again, even after the students knew exactly where each branch led them.
- Try to create your own interactive text adventures. You might surprise yourself with the excitement and sense of satisfaction that results!
- You can write directly to me for complete listings (including one possible resolution for including graphics) at:

Prince Hall School
 Godfrey and Gratz Avenue
 Philadelphia, Pennsylvania 19141
 CompuServe: 76146,306

TREE DIAGRAM of THE ATTIC ADVENTURE



Logo Connections

Software Drivers: Electronic Glue by Glen Bull and Gina Bull

Languages follow an evolutionary path. They improve or die (except for FORTRAN). The BASICs of today are nothing like the Altair BASIC that first appeared on a microcomputer. The first BASIC we used fit in 4K (kilobytes) of memory. We still have it on a paper tape. Microsoft QuickBASIC and Borland's TurboBASIC have forced each other to improve as they have quickly moved through versions 1, 2, 3, and 4, with version 5.0 on the way.

The same has been true for Logo. Logo Computer Systems, Inc. (LCSI) developed Apple Logo, followed by Apple Logo II, followed by LCSI Logo II for the Apple. Similarly, Terrapin has developed Terrapin Logo versions 1.0, 2.0, and 3.0 for the Apple, and now a Terrapin ProDOS version (Logo PLUS) is available for the Apple. Each of these new versions has been a distinct improvement over its predecessor.

The refinement and enhancement of Logo has also continued on the Macintosh. Both LCSI and Terrapin have developed versions of Logo for the Macintosh. Coral Object Logo is also an exciting version of Logo available for the Macintosh. A new version of Object Logo has been released that makes it possible to compile a Logo program and give it to someone else, even if they do not own a copy of Logo.

There has been a steady stream of improvements in the Logos available for every computer—except the IBM. At first, there were many versions of Logo for the IBM. Most of these excellent Logos, such as Digital Research's DR Logo, are no longer sold. The official IBM Logo was halted at Version 1.1 many years ago. Even though this is an excellent Logo which has held up very well over the years, it does not reflect many of the newer developments in the IBM world.

This is unfortunate, since the majority of microcomputers in the United States are either IBM or IBM compatible machines. Even though most computers in the schools are still Apples, and the Macintosh dominates desktop publishing and graphics applications, IBM compatible machines are found everywhere else. This means that a child whose family has a computer at home more than likely has access to an IBM compatible computer. The failure to develop newer versions of Logo for the IBM both hampers the teacher who wishes to recommend an educational language for home use, and undermines the claims of Logo as a viable language.

That is why we were very pleased to see the Version 2.0 of LogoWriter issued for the IBM world as well as for Apple. The Apple version of 2.0 was available first, but Version 2.0 for the IBM did not lag far behind. As we reported in last month's column, LogoWriter 2.0 makes possible the interchange of programs between the IBM and the Apple, via using text files created with LOADTEXT and SAVETEXT commands. At that time we did not actually have the new IBM version, but we now have it in our hands as we type this. (Actually, one of us is holding the manual while the other is typing.)

We are pleased to report that at first glance, LogoWriter 2.0 for the IBM appears to be very good. The new version has many desirable features, such as the ability to access subdirectories and external files. The new version also takes advantage of features in the new IBM Personal System 2 (PS/2) series, such as Models 25 and 30. On these machines, Version 2.0 of IBM LogoWriter can display (count 'em) 256 different colors. That's more colors than we can name.

You may *think* that you don't need 256 different colors, or that they serve no educational purpose. However, that's only because you haven't seen them. They are simply glorious. How many times have you attempted to convince a third-grader that she didn't really need red or yellow for a picture, and that orange was an adequate substitute for either? Dozens of colors open the door to psychological experiments, artistic discovery, and sheer playfulness. Combined with a very good FILL command, ornate stained glass designs are possible. It is possible to change the color of the turtle, the line with which it draws, or the background. Of course, LogoWriter 2.0 can only take advantage of capabilities present in a computer display, and cannot add new ones. Therefore, on older IBM models, such as the IBM XT and AT, it will only be possible to see four colors at a time, just as before. (It is possible to upgrade older models with Enhanced Graphics Adapter capabilities, known as EGA graphics, so they can display more colors.)

Charting the Colors

LogoWriter 2.0 encourages the interchange of programs between the IBM and the Apple, via exchange of text files created with LOADTEXT and SAVETEXT commands. (You will need a telecommunications package or a computer equipped with a special card to do this. See the November 1988 "Logo Connections" column for an overview of the process.) However, if you are transferring a LogoWriter program for an IBMPS/2 computer to an Apple, there is an immediate and obvious problem. Let's compare the colors: Color 2 is green on the Apple, but turquoise on the IBM Personal System 2 computer.