Logo Teaching:
A Marriage of Theory and Practice

by Ceil Humphreys

"In addition to learning Logo in this course, we will also examine several psychological theories of learning and cognition that have dominated program and curricular development in mathematics and science teaching."

A collective groan arose from the class. When I had signed up for Dr. Gary Tubb's graduate course, "Programming Languages for Education: Logo," all I wanted to do was learn more about Logo. The course was part of my masters program at the University of South Florida in Tampa. I hoped that I would be training other teachers in Logo someday.

What? More Learning Theory?

The unexpected inclusion of the learning theories was a real shock. I already had a degree in education, and several years of teaching experience. I had developed a few theories of my own; did I really need any more? Somewhat reluctantly, I accepted the theoretical aspects of this so-called "Logo" course.

We were scheduled to study Gagne, Bruner, Ausabel, Papert, and Piaget, in addition to learning IBM Logo. Each student was required to write a paper to show the use of Logo to teach a particular concept and to explain in detail the learning theory concepts selected and employed.

It wasn't until later that I discovered how important the learning theory part of the course would be in developing practical applications as a Logo teacher trainer.

One Year Later

The following year, I was given the opportunity to teach the Logo section of an introduction to computers class for undergraduate education majors. I had to cover many Logo concepts in a short amount of time,

and still keep the students enthusiastic. I wasn't sure at all how I was going to do this.

After introducing the concept of procedures, for example, I needed an effective activity to reinforce the idea and give an opportunity for practice. The students needed to understand how to organize a block structure and to discover the advantages of dividing a large programming task into smaller subtasks. I wanted the activity to be fun, open-ended, relevant, and also give the students the opportunity to be creative. Finally, I wanted the students to produce hard copies of their projects for a classroom display.

Since it was October at the time, Halloween was on everyone's minds. I found an activity in *Teaching and Computers* magazine ("Carve Logo Pumpkins," October 1985, page 50) which I could modify to incorporate all of my learning objectives.

I gave the students procedures to draw a pumpkin body, a stem, and a triangle. Then I asked them to "carve" their own jack-o'-lanterns by using these Logo tools and writing other procedures for the eyes, nose, and mouth. I encouraged them to be creative with the color commands, and to write a superprocedure to organize the entire project.

Logo Teaching continued

Surprise, Surprise

It was at about this point I realized I had applied David Ausabel's concept of "advanced organizers" to my Logo teaching! An advanced organizer, I remembered from my Logo class work, is an organizing statement which links what the learner already knows with what is to be learned, and creates an organization or structure within which the new learning will be embedded. (See Ausabel, D. P. *The Psychology of Meaningful Verbal Learning.* New York: Grune & Stratton, 1963.)

The tool procedures I furnished the students served as a link to what they already knew, and the project assignment created the basis for the new learning specified by my additional instructional objectives.

As I reviewed what I had learned about Ausabel's theory in Dr. Tubb's Logo course, I became aware of many other possible applications at every level of the curriculum. It all seemed to "click," now that I had an applied Logo perspective. Suddenly, my Logo training design took on a whole new meaning!

By the way, when my students brought in their Logo jack-o'-lantern projects, I was amazed at the wide range of "personalities" of the pumpkins. Several students commented that the tool procedures and handout I had furnished were most helpful in giving them a clear starting point. It seemed that the advanced organizers certainly had done the job!

Yes, More Learning Theory

Although I learned about the various learning theories reluctantly in my Logo course, I now feel they have greatly improved my Logo teaching methods.

If you are a teacher trainer, I encourage you to consider adding more learning theory to your Logo training. This might be an excellent opportunity to team teach with a "more learned" colleague who has not yet investigated the possibilities of Logo.

I encourage all NLX readers to investigate Ausabel (and all the other important learning theorists) and explore the richness of their work from your Logo perspectives. This could add a new dimension to your Logo teaching, as it did to mine.

Ceil Humphreys is involved with Logo teacher training at Claywell Elementary School in Tampa, FL.

From The Editor
by Tom Lough

"In your presentation, you mentioned the NLX. What is the NLX? Do you recommend that I join? How can I get in touch with them?"

Have you ever been asked these questions? I hope so. How have you answered them?

The primary objective of the National Logo Exchange is to serve teachers who use Logo in their classrooms. We do this mainly through our newsletter. In the past three and one-half years, we have provided the Logo community with more than two hundred and fifty articles for this purpose. From your many letters of support and encouragement, it seems that we are meeting many of your needs.

Many citations now refer to the NLX. The newsletter appears on practically every Logo resource list. Yet, teachers sometimes do not know about the newsletter and have difficulty learning of our address.

In order for us to be effective and to serve the Logo teachers who need new ideas and up-to-date information, our address should be readily available. When teachers ask questions such as those above, informative answers should be furnished.

NLX readers are members of a very special family. You believe in the NLX and support its activities through your subscriptions. We appreciate that support very much. Would you be willing to help provide information about the NLX to others?

If you are giving a Logo presentation or putting on a Logo workshop, tell your attendees about the NLX. If you are teaching an educational computing course which includes Logo, consider informing your students about the NLX. If you are going to an educational computing conference, think about telling those you meet of the NLX.

In addition to telling others about the newsletter, you might want to augment your Logo handouts with NLX brochures. Make them available at your Logo presentations. Give them to your workshop attendees and to those in your Logo courses. Take them to conferences and place them on the "freebie" tables.

By helping to "spread the word" about the NLX, you would also be helping bring useful Logo teaching information to those who need it.

To order a free supply of NLX brochures, write to: *NLX Brochures, PO Box 5341, Charlottesville, VA 22905.* Tell us the number of brochures requested, the date needed, and a *brief* description of the activity at which they will be distributed. If you live in the USA, please provide a street address for UPS delivery.

It is always fun to tell others about a source of information which has been useful. We hope that you feel that way about the NLX, and will enjoy making NLX information available to colleagues in your own Logo community.

Then, when someone asks you about the NLX, you can give them a brochure ... and a smile!

Thanks for helping Logo and the NLX move ...

FD 100!

Tipps for Teachers
by Steve Tipps

Getting to Know You

Talking to other people and finding out things about them is an everyday human activity. When you meet someone for the first time, you usually ask about his or her name, home, occupation, and hobby. Logo can be used to simulate getting to know someone. In the process of converting human-to-human conversations to the computer, ideas about language, Logo, and human nature are explored.

Simple Questions

Have the children consider a situation in which they meet somebody for the first time. List the many things which they might ask. Writing the questions in Logo takes only a PRINT statement.

```
PRINT [ WHAT IS YOUR NAME? ]
PRINT [ WHERE DO YOU LIVE? ]
PRINT [ WHAT DO YOU LIKE TO EAT? ]
```

The questions are not much good, however, without answers to them. When you are asking a question of another person, you stop and wait for an answer. You listen with your ears and remember with your brain. Logo can also stop, wait, listen, and remember, using the command READLIST (or REQUEST). This command causes Logo to stop and wait for another person to type in a response to a question.

If you type the commands in by themselves, you may think that the computer has died. All you see is a blinking cursor. Logo is waiting for you to type something (anything!) on the keyboard. The cursor blinks until you type and / or press <return>. You need to know what to type. When you use READLIST (or REQUEST), be sure to make the program friendly by including instructions or directions to go along with them. Make a procedure which combines the friendly question wih the request for the answer.

```
TO ASK.NAME
PRINT [ WHAT IS YOUR NAME? ]
PRINT READLIST     (for MIT Logos, use PRINT REQUEST)
END
```

You have just the beginning of a conversation. The procedure asks the question and stops, waits, and listens for an answer. The difficulty is that the procedure only repeats your name.

The computer asks, "What is your name?"

If you type "Jamie," the computer also types "Jamie."

If you ask ten other questions, the computer just repeats your answer each time. The computer echoes your answer. This habit is irritating. In addition, the computer does not remember your name or address or favorite team or anything else you tell it. The program has no way to remember your answer. As the Scarecrow, it doesn't have a brain for remembering.

Making Boxes

The lack of a brain can be remedied by creating a place to hold on to the answer with a MAKE statement using READLIST (or REQUEST).

```
TO ASK.NAME
PRINT [ WHAT IS YOUR NAME? ]
MAKE "NAME READLIST   (MIT: MAKE "NAME REQUEST)
END
```

The effect of a MAKE statement is to create a place to put the answer. Many people like to think of this place as a box with a label on it. In this case, you have created a box with the label "NAME" on the outside. The box holds the name which was typed in with READLIST (or REQUEST).
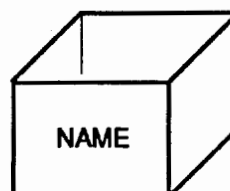
Only one name can be in the box at any time. The last name typed in is the one which is in the box. So, every time you ASK.NAME, a new name goes in the "NAME box.

Careful Questioning

This one simple question form can be adapted to any question you want to ask. But you have to be careful that your question asks exactly what you want to know. Consider a procedure which wants to know which town you live in.

Tipps continued

```
TO ASK.TOWN
PRINT [ WHERE DO YOU LIVE? ]
MAKE "TOWN READLIST  (MIT: MAKE "TOWN REQUEST)
END
```

The answers you might get include "444 Main Street," "La Vista Apartments," "California," or "around the corner." To get precise information, ask precise questions. Make up a series of procedures each of which asks a single question and gets the answer. Each question involves a seperate procedure or module. Each procedure should do one task well.

Getting Nosy

All the procedures can be combined into a single program called NOSY which asks many questions of people.

```
TO NOSY
ASK.NAME
ASK.TOWN
ASK.STREET
ASK.FAVORITE.FOOD
ASK.FAVORITE.TEAM
ASK.BEST.VACATION
END
```

Students invent many, many questions depending on their interests. College students in an Introduction to Computer Science class worked up an elaborate NOSY program to screen prospective dates with such personal questions as kind of car, income, looks, and favorites of every kind. By working on separate procedures which did only one thing well, they were developing modular programming habits.

What's in the Box

Each of the ASK procedures makes a box and collects an answer in it. The procedure now has a brain, but needs a way to retrieve what it remembers. You may want to know what THING is in the "NAME box or what THING is in the "FAVORITE.FOOD box. In Logo, you PRINT the THING in the box.

?PRINT THING "NAME
JAMIE
?PRINT THING "FAVORITE.FOOD
BURGERS

A shorthand way of finding out what is in the boxes is the more commonly seen :NAME (with a colon, or "dots") instead of THING "NAME.

?PRINT :NAME
JAMIE

?PRINT :FAVORITE.FOOD
BURGERS
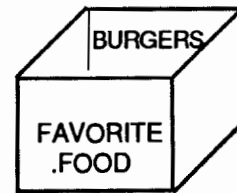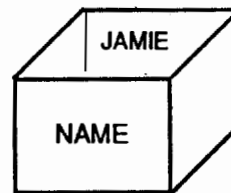
Both forms work equally well at retrieving the contents of the variable. However, our experience is that using THING as the referent to the contents of the variable box results in better understanding of what is going on. When students (or teachers) later use the short form, :NAME, they carry with them the mnemonic of "the thing in the NAME box." Without this experience, they are constantly confused about when to use "NAME and when to use :NAME.

> "NAME is the label
> :NAME is the thing in the box

In more advanced work with lists, a full understanding of THING is absolutely essential. By starting with THING, list processing is much easier.

Tell All

After asking all the questions and remembering all the answers, Logo is ready for a procedure to tell all about a person. You could just summarize the facts with simple PRINT procedures.

```
TO TELL.ALL
PRINT THING "NAME
PRINT THING "TOWN
PRINT THING "FAVORITE.FOOD
...
END
```

The difficulty is that the procedure prints out just the information and no description of what it is. A better way of doing this is with a SENTENCE which tells what is being printed.

```
TO TELL.ALL
PRINT SE [ YOUR NAME IS ] THING "NAME
PRINT SE [ YOUR TOWN IS ] THING "TOWN
PRINT SE [ YOUR FAVORITE FOOD IS ] THING "FAVORITE.FOOD
...
END
```

SE is the short form for the SENTENCE command. It is useful when you are going to print two things on the same line.

Tipps continued

Interviewer Extraordinaire

These questioning tools can be used in many ways in the classroom. One way is to have the student of the week answer the questions. Then turn on the printer to share important information with everyone in the class. Another use would be in language or social studies for collecting information about famous people. You could write an interview for George Washington, Laura Ingalls Wilder, or Ralph Nader. Fictional and fantasy characters such as Winnie the Pooh or E.T. might also be interviewed. Interviewing could be a creative writing stimulus. Older students might study interviewing techniques on TV. Remember the question, "If you could by any kind of tree (or animal or whatever), what would it be?"

At the end of the interview, students can also save all the information. One simple way is to save all the procedures and the current answers in a file.

?SAVE "GEORGE.WASHINGTON

Of course, all the procedures are repeated in every file which is handy but can use a lot of disk space. Another method is to erase the procedures using ERPS (or ERASE PROCEDURES) before saving the file. Then only the current answers are put into the file.

From a simple question to a whole interview requires only five Logo commands: PRINT, READLIST (or REQUEST), MAKE, THING, and SENTENCE. They are simple commands which lead to exciting possibilities in writing and reading. The real goal of getting to know you is working and playing with language just as the real goal of Logo is expressing ideas.

If you have additional ideas to share, please send them to Steve Tipps, Midwestern State University, 3400 Taft Boulevard, Wichita Falls, TX 76308, or on CompuServe at 76606,1623.

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX, and has been involved with Logo since 1982. He conducts Logo workshops for school systems throughout the United States, and is a popular conference speaker.

Logo Resource Packet Available

The International Council for Computers in Education (ICCE) has for sale a Logo resource packet which contains copies of 2 Logo issues of *The Computing Teacher* magazine and reprints of nearly 20 additonal Logo articles. The packet makes an excellent supplemental reading resource for teacher training courses, and is available for $10.50 from ICCE, 1787 Agate Street, Eugene, OR 97403.

Logo Teacher Feature

by Rebecca Poplin

Featuring: Peter Anderson

Logo teachers are explorers and discoverers, always searching for new ideas, new perspectives, and new understandings. Peter Anderson, a teacher from Sharon, VT, responded to his urge to investigate the uses of computers in education, and set out last year on an exciting odyssey many of us have only imagined.

On the Road

Peter was granted a year's sabbatical from his position as English teacher at the Marion W. Cross School in Norwich, VT. He used his time to visit schools and conferences all over the United States, England, Norway, and Sweden. Peter defined three major interests related to computers in education: (1) the effect of school policy on computer use, (2) teacher training, and (3) word processing: using technology to help students write.

Peter had been using computers for Logo as well as word processing in his classroom. He was fascinated with the creative stimulation of Logo problems and wanted to learn more. His first venture was to attend Dan and Molly Watt's Logo Institute in Keene, NH. (Ed. note: For information on this year's institute, write to Tony Stavely, Keene State College, Keene, NH 03431.) There, he worked with 40 teachers from all over the country. His experience raised many questions concerning what education is about and what the role of technology should be.

Metaphors Wanted

Meanwhile, he became fascinated by Logo's use of words and lists, but had difficulty at his level of programming experience. As Peter tried to devise a metaphor to help him understand how Logo processes lists, he decided to write a play for his fifth and sixth grade students called "Logo Processes a List."

Peter spent the fall visiting Harvard and MIT, attending seminars. He worked at Coleco for two weeks on the conceptual stage of software design. Another trip was to the University of New Hampshire to work with Donald Graves, an educational researcher who is studying the use of computers to promote writing in the early grades. Peter also went to the Minnesota Educational Computing Corporation (MECC) conference in Minnesota, and found others who were interested in his ideas for Book / net, an international publishing exchange.

Teacher Feature continued

Stooper Troops

In January, with his wife, Ina, and younger son, Angus, Peter packed his camper van and began to travel all over the United States. He soon learned that the van's height made it difficult to don jacket, tie, and briefcase for his interviews. Angus quickly dubbed him "Stooper Man," and the name stuck for the rest of the trip!

The first stop was in Washington, DC, to visit with Frank Withrow at the Department of Education. Peter was encouraged to find that they both shared many of the same ideas about computers in education. Visits to schools from Maryland to South Carolina convinced Peter that many school districts are still struggling with the implementation of computers.

He decided to visit the Florida Educational Computing Conference, because Florida had just created a statewide computer curriculum. What Peter found was that teachers had been given a set of computer competencies and were desperately visiting vendors to find software for specific computer skills. Attendance at the Texas Computer Education Association conference in Austin revealed a similar situation there. Peter spent time in Houston, learning about the school district's in-house software development team, where one of many products has been a computer based "English as a Second Language" program.

In Arizona, Peter visited with some educators who were very interested in evaluating software. He became even more convinced that there is a lifetime's worth of work with a good word processor, Logo, and a data base, and became more wary of passive or non-interactive software.

A sojourn in Bruce Fleury's classroom at the Magnolia School in El Cajon, CA, provided Peter with an example of a way to decentralize computer equipment and prevent what he calls the "shrine room mentality" apparent in many schools. On a roof-covered deck with access to several classrooms, a computer / printer setup was available for use by different groups of students. Peter also observed second graders being taught by fifth graders.

While in the San Francisco area, Peter did a workshop on publishing with computers, and shared what he had seen on his travels.

Up and Away

After a time back in Vermont, Peter left the US to pursue his interest in policy in countries with a centrally organized state educational system. In Sweden, he found that the government has commissioned educational software which reflects the Swedish culture and identity. Except for a single pilot program, Sweden has not started using computers in elementary schools.

Teacher training is a major problem there, and it is a struggle to keep teachers in education. In addition, the education authorities want to study computers more extensively and proceed slowly.

In Sweden, Norway, and the United Kingdom, Peter made contacts for his Book/net project, and has since organized the participation of fifteen schools in ten different countries. Peter has published an article about his work, "Classroom Publishing with Micros," in the October 1985 issue of *The Computing Teacher.* Peter has been invited to return to Sweden in March to make a follow-up presentation.

Peter's exploration has led him to several conclusions about the state of computer education. One is that teacher training is a key problem everywhere, with no totally satisfactory solutions. He is encouraged that people no longer seem to have the notion that every child should learn BASIC, and that people are challenging the idea of the concrete curriculum. It became even more obvious to him that there are no easy solutions. We must see technology in education as an experiment, and be prepared to adapt as we learn more.

Send Your Cards and Letters

Peter would be interested in further contact with NLX readers concerning the Book/net, his travel experiences, workshops, or his Logo play.

Peter Anderson
Route 1, Box 444
Sharon, VT
(802) 649-1519
CompuServe 72437,2251

Sharing ideas is a big part of what the NLX does for teachers. If you have ideas you would like to share through this column, or you know of teachers who should be featured, please contact:

Rebecca Poplin
2421 Fain Street
Wichita Falls, TX 76308

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

Logo Challenge Offered

Apple Computer Clubs International have announced their Challenge Program, a merit award program similar to that of scouting merit badges. One of the eleven challenge areas is Programming in Logo. This is an excellent opportunity for students to design and carry out Logo activities which will help them learn even more. For more information, write to Challenge Program, Apple Computer Clubs International, PO Box 948, Lowell, MA 01853.

NLXionary
A Lectionary of Selected Logo Readings

with Commentary and Opinion
by Griff Wigley

"The Rhetoric and Reality of Expanding Educational Choices," by Joe Nathan, *Phi Delta Kappan*, March 1985.

"The Synthesis of Research on Schools of Choice," by Mary Anne Raywid, *Educational Leadership*, April 1984.

>I digress.

>These articles have nothing to do with Logo, and yet they have everything to do with Logo. I bring them to your attention because I believe they address the philosophy of Logo. Although the subject seems far removed from everyday classroom activities, it directly affects classroom functioning in a multitude of unseen ways.

>The issue is permitting families to select from among different public schools. Why? I'll list some of my reasons. But first, think about Logo and its philosophy for student involvement. There are many components, but choice is a cornerstone. Without choice, exploration and discovery are hollow. We've all seen instances where the spirit of Logo was thoroughly undermined when implemented in a school district.

>If you look closely, lack of choice is usually at the center. It may have been that teachers had no choice about their participation in Logo's implementation. Possibly students were expected to learn Logo according to a prescribed scope and sequence - usually devoid of choice. Successful Logo project implementations almost by definition involve choice.

>But the structure of public education in America, land of freedom, uncharacteristically defines the parents' "choice" of where to send their children to public school as a function of where the parents happen to live or are able to live.

>This is the perfect recipe for creating uninvolved and unmotivated parents, just as in the schools for creating uninvolved and unmotivated students.

>The research indicates that when families are permitted to select from among different public schools:

• parental satisfaction with the educational system increases

• students' self-concept, behavior, attitudes toward learning and towards schools improve

• student attendance and achievement increases

• teacher morale goes up as teachers have opportunities to work in programs which fit their educational philosophy (Teachers also like the opportunity to use special expertise they've developed. One way to help make this happen is to provide faculty with opportunities to create programs and then permit families to select from among them.)

• community support for schools increases as people hear about diverse high quality offerings

• in the beginning, affluent families have more information about choices; however, as times goes on (and the rules stay the same), the difference in knowledge about choices decreases

• although "choice" can be misused by a district to increase segregation, many are now using it to increase integration.

>As this issue continues to gain national focus, I urge you to read about it and discuss it, keeping your Logo philosophy in mind.

If you would like to respond to these articles and commentary, or know of other Logo related articles and issues which may be of interest to our NLX readers, please write to: Griff Wigley, 918 College Street, Northfield, MN 55057, or send electronic mail to me on CompuServe at 72007,24.

Griff Wigley is a co-owner of Family Computing, Inc., a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

Ohio Logo Project Going Strong

"Integrating Logo into the Classroom" is the theme of a year-long project being undertaken by the Educational Computer Consortium of Ohio (ECCO) in 1985 - 86. With the help of a grant from the Martha Holden Jennings Foundation, several workshops and two Logo conferences are being held.

During this project, six of the hands-on workshops present Logo materials that may be taken back to the classroom. Topics includs elementary math and science, elementary language arts and social studies, secondary math and science, and secondary language arts. For those teachers who are looking to move further into the use of Logo in the curriculum, Michael Tempel of Logo Computer Systems, Inc., is presenting three workshops which explore the topics of language, math, and science projects with Logo.

The spring Logo conference of 1985 was the starting point and the 1986 spring conference will wrap up the project. ECCO has provided opportunities for the beginning Logo user and materials for teachers who do not have time to develop their own. The project is an exciting venture for all who participate.

For more information, write to Josie Drushal, Elementary Computer Resource Teacher, Wooster City Schools, Wooster, OH 44691.

Listful Thinking
by Glen Bull and Paula Cochran

Story Time

?NLX.STORY.STARTER

Once upon a time MRS. JONES was thinking about how to integrate what her students were studying in GRAMMAR into their WRITING assignments. She was casually leafing through the latest NLX, when suddenly . . .

A story starter, such as the one above, is supposed to get the creative juices flowing. It should give the reader / writer some ideas, and a basis for continuing the story in his own way. Naturally, we expect that the story above will have a happy ending. In this month's column we will discuss how to use Logo to develop story activities and story starters geared toward specific language arts goals.

Getting Started

The Logo-generated Christmas story below is one which we recently received from a local elementary class. The class is using Apple Logo II, which allows upper and lower case text. Some words in the story are randomly chosen (shown capitalized), making the story slightly different each time it is printed on the screen (or printer). Our thanks to Cheryl Wissick and her students in the Albemarle County Public Schools for sharing this story.

> There were 14 BUSY elves at the North Pole.
> SUSAN was wrapping the TRUCKS and
> GEORGE was packing the COLD sleigh.
> Then Santa found LAZY ROY sleeping behind
> the boxes of BIKES. Santa was glad the other
> CHEERFUL elves were hard workers, so he and
> SALLY treated everyone to HOT CHOCOLATE.

The students who wrote this story worked as a group to think of key words for the story, such as the toys, the names of the elves, and the adjectives which describe them. The Logo story procedure is picking from lists of these words.

Talking about the lists of words caused students to review some of their recent work on parts of speech. This will be discussed in more detail below.

The students got excited about the stories for two reasons: they enjoyed seeing what funny and interesting combinations happened each time the story procedure was called, and they enjoyed seeing their own suggestions appear in the stories. They were encouraged to use the stories as story starters, writing their own ideas about what might have happened next. The results were so successful that they used them in a holiday program for their school.

Before we get into how such a story procedure and some enhancements might work, let's look at a very simple story starter procedure.

Start at the Very Beginning

Students of all ages seem to enjoy seeing their names and the names of their friends in stories. We will need a PERSON procedure, which uses PICK. (If you are using Version 1 of Terrapin Logo, you will also need to type in the ITEM and COUNT procedures found at the end of the column.)

```
TO PICK :LIST
OP ITEM 1 + (RANDOM COUNT :LIST) :LIST
END
```

PICK is a general-purpose tool widely used within the Logo community, and we have employed it in a number of our projects. PICK is the only tool necessary for a basic story starter, such as we described in the November, 1984 Listful Thinking column, "Count Your Blessings" (NLX 3 (3), pp. 7-10).

```
TO PERSON
OUTPUT PICK [ SALLY TOM GEORGE SUSAN ROY BEN ]
END
```

This procedure picks randomly from the list of names we provided:

?PRINT PERSON
GEORGE

It is easy to incorporate PERSON and other similar procedures into a story starter. Using PERSON as a model, we could add an ANIMAL procedure for a story about a class trip to the zoo.

```
TO ANIMAL
OUTPUT PICK [ GORILLA PARROT LION ELEPHANT MONKEY SEAL ]
END
```

```
TO ZOO.STORY
PRINT [ Mrs. Jones' fifth grade class went to the zoo. ]
```

```
PRINT (SE PERSON [made] ANIMAL [noises on the bus.])
PRINT (SE PERSON [was hoping to see a real] ANIMAL [.])
PRINT [ ]
PRINT (SE [The class was walking by the] ANIMAL [cage])
PRINT (SE [when they heard a] ANIMAL [.])
PRINT (SE [Suddenly,] PERSON [looked up and saw . . .])
END
```

Depending on which words are PICKed, you might get the following story starter when ZOO.STORY is called:

> Mrs. Jones' fifth grade class went to the zoo.
> BEN made LION noises on the bus.
> GEORGE was hoping to see a real GORILLA.
>
> The class was walking by the ELEPHANT cage
> when they heard a MONKEY.
> Suddenly, SUSAN looked up and saw . . .

Any relevant class project or seasonal event can be used as the basis for a story starter. If only one story starter is needed, it is hard to see much advantage in using Logo for the task instead of a regular word processing program. The differences are apparent, however, when the teacher prints out 20 story starters, all with the same basic structure but each slightly different. Using this method, the teacher can easily print out a unique story starter for each student in the class.

Stories and Grammar

How can Mrs. Jones use the Logo story starter template to emphasize grammar goals, too? We strongly encourage teachers to involve the students in the design of the story starter, and not just use the finished product for a writing exercise. Depending on the age and Logo background of the students, this can be done on a group or individual basis.

The ideal situation is for a teacher to be working with a small group of students, using a large monitor or projector so that everyone is looking at the same screen. Then together they can work on making lists of words that will produce interesting stories.

Students do not necessarily need to know how the story starter procedure works, or even what it looks like. The teacher should prepare the skeleton of the story ahead of time, and get the lists of key words started with a *few* good examples. As the activity begins, the teacher explains the topic of the story, and shows the students some examples of the story she has made so far.

After they talk about the stories on the screen, she asks them to help her think of more words for the story. She edits the appropriate procedure (PERSON, ANIMAL, etc.) and adds their suggestions for words that match that category and that would sound good in the story.

```
TO ANIMAL
OP PICK [ GIRAFFE SNAKE_____ ]
END
```

They should go back frequently to look at whole stories on the screen, to see if the new words do fit in the story (and to see the interesting results!).

The story starter should be written to emphasize the parts of speech that the teacher wants the children to think about and discuss. For example, the Elf Story presented earlier emphasized nouns and adjectives. That is, the words which changed in the story and which the students discussed most were nouns (trucks, bikes, hot chocolate, etc.) and adjectives (cheerful, lazy, cold, busy, etc.).

The same story could have easily been used to work on verbs and adverbs. Even the names of the procedures can emphasize recent grammar lessons. With apologies to the authors, here is a possible variation of their Elf Story, revised to make it a lesson on verbs and adverbs.

```
TO ADVERB
OP PICK [ HAPPILY QUICKLY WILLINGLY NOISILY ]
END

TO TRANSITIVE.VERB
OP PICK [ PACKING WRAPPING WASHING FIXING TYING ]
END

TO INTRANSITIVE.VERB
OP PICK [ SLEEPING WORKING THINKING SNORING
RESTING CRYING LAUGHING ]
END
```

Using the new lists, here is what the Elf Story might look like:

> There were 14 elves working HAPPILY
> at the North Pole. SUSAN was
> WRAPPING the gifts and
> GEORGE was PACKING the sleigh.
> Then Santa found ROY SLEEPING
> behind the boxes of new bicycles.
> Santa knew that the other elves had been
> WORKING, so he and SALLY QUICKLY
> treated everyone to milk and cookies.

The story starter procedure can be written like ZOO.STORY, using PRINT statements and SENTENCE to combine the elements of each phrase in the story. This is the most straightforward way to write the story template, but typing PRINT, etc., can become a nuisance if you use this and other text-heavy applications frequently. This can be remedied and YOU can enhance the simple story starter through the use of a few Logo text tools described below.

Text Tools

Last month, you recall, we outlined the concept of a list machine. A list machine is a basic template or model for a procedure which operates on a list. With a few slight changes, the basic list machine can be modified to perform a variety of other tasks.

INSPECT is a basic Model-T list processing machine that we have discussed in previous columns. A slight modification to INSPECT will change it into a story - printing machine.

```
TO INSPECT :LIST
IF EMPTYP :LIST [ STOP ]
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

Listful Thinking continued

If you are using Terrapin Logo, use the following version of INSPECT.

```
TO INSPECT :LIST
IF :LIST = [ ] STOP
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

INSPECT takes a list of items, and prints each item on a separate line.

```
?INSPECT [A B C]
A
B
C
```

If we enter text into a procedure without putting in any PRINT statements, we can still use INSPECT to print the text on the screen later. For example, if we start a story like this:

```
TO SAMPLE.STORY.TXT
MRS. JONES' FIFTH GRADE CLASS WENT TO THE ZOO.
EVERYONE HAD A GOOD TIME.
END
```

we can then print the story like this:

```
?INSPECT TEXT "SAMPLE.STORY.TXT
```

MRS. JONES' FIFTH GRADE CLASS WENT TO THE ZOO.
EVERYONE HAD A GOOD TIME.

This eliminates the need to type in PRINT statements and brackets. (We have adopted the convention of adding the .TXT suffix to procedure names when the procedures consist of text and will not run correctly alone.)

Why can't we use INSPECT for our story starters, without modifying it? It is because we have procedures like PERSON, ANIMAL, etc., embedded in the text. ZOO.STORY contained the SENTENCE command and parentheses, in order to have the variable words and sentence pieces print out together. The first two lines of the ZOO.STORY procedure looked like this:

```
TO ZOO.STORY
PRINT [Mrs. Jones' fifth grade class went to the zoo.]
PRINT (SE PERSON [made] ANIMAL [noises on the bus.])
. . .
```

If we omit the PRINT statements, SENTENCE, and the parentheses, how will Logo know the difference between regular text and embedded procedure names?

Our Contest Entry

Two small changes in INSPECT produce a new procedure which can be used to print a story. (If we were not disqualified, this would be our entry in the NLX List Contest. See the end of the column for details about the contest, and for the Terrapin version of FORMAT.)

```
TO FORMAT :STORY
IF EMPTYP :STORY [PRINT [ ] STOP]
PRINT FORM.LINE FIRST :STORY
FORMAT BUTFIRST :STORY
END
```

(The boldface in the FORMAT procedure indicates the changes made from INSPECT. For comparision, here is the INSPECT procedure once more.)

```
TO INSPECT :LIST
IF EMPTYP :LIST [ STOP ]
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

. The procedure FORM.LINE is placed after PRINT in INSPECT, and is used to form each line of our story. (For now, don't worry about how it does this. Just type it in.)

```
TO FORM.LINE :LIST
OUTPUT RUN FPUT "( FPUT "SE LPUT ") :LIST
END
```

Now we can leave out all the PRINT statements when we enter the text of our story. Create a procedure with the following text. (If your version of Logo does not permit use of lower case, enter the text in all capital letters.)

```
TO ELF.STORY.TXT
[There were 14 elves working] ADVERB
[at the North Pole.] PERSON [was]
TRANSITIVE.VERB [the gifts and]
PERSON [was] TRANSITIVE.VERB [the sleigh.]
[Then Santa found] PERSON INTRANSITIVE.VERB
[behind the boxes of new bicycles.]
[Santa knew that the other elves had been]
INTRANSITIVE.VERB [, so he and ] PERSON ADVERB
[treated everyone to milk and cookies.]
END
```

To print ELF.STORY.TXT on the screen, type:

```
?FORMAT BF TEXT "ELF.STORY.TXT
```

Obviously we don't want to bother typing this each time, so we made a procedure to do it:

```
TO DISPLAY :PROCEDURE
FORMAT BF TEXT :PROCEDURE
END
```

Once these procedures are in place, you need only type:

```
?DISPLAY "ELF.STORY.TXT
There were 14 elves working QUICKLY
at the North Pole. ROY was
PACKING the gifts and
GEORGE was WASHING the sleigh.
Then Santa found SUSAN RESTING
behind the boxes of new bicycles.
Santa knew that the other elves had been
WORKING, so he and BEN HAPPILY
treated everyone to milk and cookies.
```

This use of DISPLAY and FORMAT saves the hassle of many PRINT statements, etc., not only for the teacher, but also for students assigned to work independently on stories or story starters. They will be much less likely to be bothered with Logo syntax details such as missing parentheses. This allows them to focus their attention on the English goals of the exercise.

Listful Thinking continued

Of course, the tools should be provided by the teacher on disk, along with a model of the inside of a STORY.TXT procedure. The only thing students will need to know to make it work, besides how to edit a procedure, is to type DISPLAY "STORY.TXT.

How *Does* FORMAT Work?

Earlier we asked you not to worry about how FORMAT works. We just wanted you to type it in and use it. However, we know that there are a few of you who will worry anyway. If you are one of those persons, here's an explanation. (If you're not interested, skip ahead to the next section. You won't hurt our feelings!)

FORMAT is a straightforward variation of INSPECT. The primary difference is that the middle line (PRINT FIRST :LIST) becomes (PRINT FORM.LINE FIRST :LIST). The FORM.LINE procedure takes a line of the story such as this:

[[There were 14 elves working] ADVERB]

FORM.LINE inserts the command to form a sentence (SENTENCE, or SE for short) and places parentheses around the text. This turns the list into a valid line of Logo code.

[(SE [There were 14 elves working] ADVERB)]

FORM.LINE runs the code, by using the command RUN, to form a sentence. Lisp is the only other higher level computer language we know of that can do this.

Mixed Parts
(Parts Is Parts)

Now that DISPLAY, FORMAT, and FORM.LINE have removed some of the overhead of story starters and stories which use word lists, let's look at a couple more ways to use them in grammatically oriented activities.

The class which sent us the original ELF.STORY did an interesting exercise using the story template alone. Instead of PERSON, ADVERB, etc., those places in the story were replaced with blanks. Each student filled in the blanks on a worksheet with his own words, and then compared them to the stories produced with the word list procedures.

Using the story with blanks could be either a pre-computer activity, to get everyone thinking before the group activity, or a follow-up activity to carry over the use of certain parts of speech in a writing exercise.

Another possibility for a follow-up activity involves scrambling the story a bit. Suppose Logo generates a story with key words in the wrong places? Naturally, the key words are verbs or adverbs, or another part of speech which the class is studying.

Let's scramble ELF.STORY.TXT, by just moving INTRANSITIVE.VERB and TRANSITIVE.VERB around:

```
TO ELF.STORY.TXT
[There were 14 elves working] ADVERB
[at the North Pole.] PERSON [was]
INTRANSITIVE.VERB [the gifts and] PERSON
[was] INTRANSITIVE.VERB [the sleigh.]
```

```
[Then Santa found] PERSON TRANSITIVE.VERB
[behind the boxes of new bicycles.]
[Santa knew that the other elves had been]
TRANSITIVE.VERB [, so he and ] PERSON ADVERB
[treated everyone to milk and cookies.]
END
```

Now when we print the story, it could come out like:

?DISPLAY "ELF.STORY.TXT
There were 14 elves working QUICKLY
at the North Pole. ROY was
SNORING the gifts and
GEORGE was THINKING the sleigh.
Then Santa found SUSAN TYING
behind the boxes of new bicycles.
Santa knew that the other elves had been
FIXING, so he and BEN HAPPILY
treated everyone to milk and cookies.

We could have scrambled any elements of the story, but the contrast between verbs which do and do not take objects would be a good place to start in some classes. Using a printout, each student should unscramble the story he received, using the same words in different places. Starting with familiar stories they have already thought about, students can work up to repairing the syntax in stories which are new and have more than one set of mixed parts.

The teacher can use class events and current topics of interest as the basis for the stories, reinforcing not only grammar in writing activities, but also new vocabulary from social studies or science class.

Good luck, Mrs. Jones, wherever you are!

The Contest!

A Koala Pad will be awarded for the best procedure which makes use of the fact that a procedure is just a series of lists. Submissions must be received by February 28, 1986. Send your procedure listings to Glen Bull and Paula Cochran, University of Virginia Curry School of Education, 109 New Cabell Hall, Charlottesville, VA 22903.

In addition, if you have other list processing activities you would like to share with us, please send them to the above address, or CompuServe 72477,1637.

Procedures for Terrapin Users

```
TO FORMAT :STORY
IF :LIST = [] PRINT [] STOP
PRINT FORM.LINE FIRST :STORY
FORMAT BF :STORY
END
```

Procedures for Terrapin, Version 1 Only

We use the Conan (hack and cut) approach to programming. Our COUNT and ITEM procedures published previously called other (recursive) procedures which did the work. Alison Birch wrote to point out that COUNT and ITEM can be (and typically are) written more efficiently by making full use of recursion within single procedures. These compact

Listful Thinking concluded

recursive versions follow. (Similar versions are found on the disk, Terrapin Utilities II, which can be obtained from Terrapin Inc., 222 Third Street, Cambridge, MA 02142, tel. (617) 492-8816.)

```
TO COUNT :OBJECT
IF EMPTY? :OBJECT OP 0
OP 1 + COUNT BF :OBJECT
END

TO ITEM :NUMBER :OBJECT
IF :NUMBER = 1 OP FIRST :OBJECT
OP ITEM :NUMBER - 1 BF :OBJECT
END

TO EMPTY? :OBJECT
OP ANYOF :OBJECT = [ ] :OBJECT = "
END
```

Alison also sent a sample issue of her newsletter, *Kaleidoscopes*, which we found to be excellent. For more information, write to Kaleidoscopes, Computers for a New Education, 57 Waverly Street, Suite 100, Belmont, MA 02178.

Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. Paula Cochran is finishing her doctorate at the Curry School of Education at the University of Virginia. She is interested in Logo applications in language arts and speech-language pathology.

TurtleTips

by Jane Toth
and Donna Lanyi

Our *hearts* go out to students who are "going around in circles" trying to figure out the CIRCLE and ARC commands. After working with Logo for three years, we still struggle with choosing the correct numbers. We have finally figured out some basic steps that helped us conquer our "arcphobia."

The Turtle Teacher

One of the first concepts children have to understand is that circles and arcs on the computer are really a series of connected straight lines with turns in between. To demonstrate this idea, take your class to the gym and use the painted circles on the floor. Have all students use their own feet as straight line segments. Have them walk around the circles heel-to-toe to find out how many of their straight lines it takes to go around once. Discuss with your class why students' numbers differ (size of circle, length of straight segment, size of foot, etc.). An effective manipulative device to illustrate this

same idea is the use of straw sets. Create straw sets by cutting regular drinking straws to various lengths. For example, make a set of one-half inch long white straws, a set of one inch long striped straws, and so on. Have students connect a given number of straws of a particular set to create a circle. They could also construct straw circles by threading a string through a number of straws and then curling the string to form a circle. How big is the white circle? How about the striped one?

Next, have students place straws on given circle patterns. How many white ones were needed? How many striped ones? Again discuss the importance of the length of the straws and the amount of turn between each straw. (Ed. note: For additional related activities, see "Tipps for Teachers: Wonderful Wiggles," in the January 1985 NLX.)

Next, we give our students two separate worksheets, one for arcs and one for circles each with a set of given commands. Students type in the commands and copy the shapes from the monitor onto their worksheets. We have found that students keep these sheets as references for using circles and arcs in other procedures.

Ready for a challenge? Elizabeth Moody from Trinity School in New York gave us worksheets with the following shapes. She suggested giving students the first few commands and challenging them to finish the designs.

```
TO SNAIL
REPEAT 30 [ RT 6 FD 1 ]
REPEAT 30 [ RT 6 FD 2 ]
...
```

```
TO CONE
RT 60
REPEAT 30 [ FD 4 RT 2 ]
...
```

Procedure of the Month

How's your "heartwork?" Have students put their knowledge of arcs and circles to practical use. They could design heart people for bulletin boards, Valentine cards, heart mobiles, and Valentine stories with their heart people as characters. Students could even turn their stories into a Valentine puppet play using their heart people on popsicle sticks as the characters. If you are interested in a copy of our heart person procedure, please send us a self-addressed stamped envelope.

TurtleTips continued
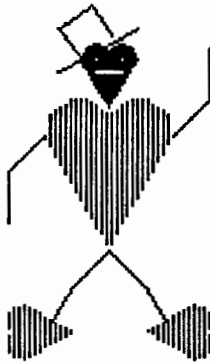
We invite any teachers or elementary students to send Logo procedures or ideas to be shared in this NLX column. Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691, or Compuserve 72477,2623.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

LIFT's

Teacher to Teacher
A Review of
New Logo Materials

by James Fry

This month's column features a book which represents the next wave of Logo materials that are finally reaching the bookstores: books on teaching *with* Logo, not just teaching Logo.

Teaching with Logo - Building Blocks for Learning, by Molly and Dan Watt, 1986, published by Addison-Wesley Innovative Division, 2725 Sand Hill Road, Menlo Park, CA 94025, $16.88.

Target Audience: Anyone who teaches with Logo to students of any age.

Content: The book focuses on the learning of a group of people in a Logo community. The main focus is on a student named Lynn and how she works and responds to eight different sessions of working with Logo. *Teaching with Logo* is a unique demonstration of learning with Logo and the teaching of many "powerful ideas."

Format: The book consists of an Introduction and Guide, 9 chapters, an appendix, and an index. The introduction has an excellent discussion on Logo and and a list of 20 powerful ideas for those who use Logo. (Ed. note: See also Molly Watt's front page article in the September 1985 NLX.) Each chapter progresses to an increased level of sophistication and shows specific strategies that a teacher may use. The chapters also make comments on what has been taught and learned. Each chapter ends with examples of class records of the work done by each student.

The last two chapters follow a different pattern. The role of the teacher is discussed in more detail, and advanced Logo projects are offered.

The procedures in the book are written in Terrapin / Krell (MIT) versions of Logo. A listing of the procedures for other popular Logo versions is contained in the appendix.

Strengths: This is a book that many people need to read. A number of school districts are giving up on Logo because they have never really understood it. They have been teaching Logo and never realized that they should be teaching *with* Logo.

The conversations that have been recorded between students and teachers are valuable teaching aids. The notes and Logo class records at the end of each chapter offer many useful ideas.

Comments: Thank you, Molly and Dan! It is wonderful to see this exciting and unique guide with fresh Logo ideas. Keep up the good work!

• • • • •

What new Logo materials and books do you find useful in teaching? We invite you to send information to the Logo Information For Teachers (LIFT) group, PO Box 5396, Plymouth, MI 48170, or send electronic mail on CompuServe to 76317,565.

James Fry uses Logo with his Chapter 1 remedial mathematics students at Novi Community School, Novi, MI, and is a co-founder of the Logo Information For Teachers (LIFT) group.

Logo Institute Announced

Dan and Molly Watt and Tony Stavely have scheduled their summer Logo institute to be held at Keene State College, Keene, NH, July 14 through 27, 1986, to follow the Logo 86 conference at MIT. Past participants have spoken highly of this institute (see Logo Teacher Feature on page 5). For more information, write to Tony Stavely, Logo Institute, Keene State College, Keene, NH 03431, or call (603) 352-1909, ext. 554.

Logo Disserts:
Dissertations
Dealing wih Logo

by Barbara Elias

What are the programming abilities and thought processes of five year old children when they learn to program a computer with Logo? What problems do they encounter and how do they feel about programming? A study of five year olds by Sandra N. Hines may provide some insights for kindergarten teachers and curriculum planners.

Hines conducted a ten week study of children enrolled in a public school kindergarten class. Children were randomly selected from a group which had passed a screening test on numeral and letter recognition.

Daily programming sessions of about twenty minutes each were held in a room near the regular classroom. Games were used to introduce the computer and Logo to the children. Additional games served to help the children in orienting their bodies in space. Each child wore labels on the backs of his or her hands to reduce confusion about directionality.

Prior to programming on the computer, children drew their designs on paper. These designs were then enlarged on a drawing on the playground. Children walked along the lines, counting the number of steps needed for each part.

The researcher recorded the number of steps required; these numbers became the inputs for the computer programs. After several outdoor drawing experiences, the children could write programs to draw their pictures using only the paper drawings. A number line was used as a teaching tool for number inputs, number quantity, and comparison of numbers.

Data resulted from (a) dribble files, notes, and tape recordings of individual sessions, (b) test results from a number quantity test and scoring on nine Piagetian tasks, and (c) from an informal interview. Two tests on programming, given at the end of the study, provided additional data.

The children in this study exhibited a variety of abilities in programming. All were able to work out procedures with the researcher's assistance. Two could be considered independent programmers, one could not be considered a programmer, and two could write procedures but experienced difficulty in correcting errors.

The number of procedures completed by the participants varied, and indicated their interest and ability to attend to programming tasks. Those who completed the most procedures liked programming. Those who completed the fewest procedures preferred the games, and therefore spent less time on projects.

Task perseverance, interest level, and the use of thinking strategies instead of repetitive behavior appear to be factors in programming success. Some five year olds can program, but they need help in programming. Flexible instructional arrangements which provide this help and the use of prerequisite skill games and programs which require little typing may be good starting points for computer "literacy" at the kindergarten level.

>>>Hines, S. N. (1984). *A Qualitative Analysis of the Computer Programming Abilities and Thought Processes of Five Year Old Children* (Doctoral dissertation, North Texas State University, 1984). *Dissertation Abstracts International, 45,* 1983-A.

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of VA.

Q. I understand that some computer languages are compiled, while others are interpreted. What exactly does this mean? Is Logo compiled or interpreted?

by Jim McCauley

A. A *source* computer program is a text which can be read and understood (one hopes!) by human beings. Unfortunately, most computers don't "speak" the high level programming languages (such as Logo and Pascal) that human beings prefer to use. Each type of computer has its own unique language, called "machine code."

There are two ways of arranging for a computer to "understand" and run the text. First, the source text may be *compiled* into a sequence of machine code instructions by sending the text through a compiler program. This program attempts to translate the entire text all in one gulp, as it were.

If there are errors in the program, the compiler will either stop, print out one or many error messages, or go completely crazy, depending on how "smart" the compiler is. If there are no errors, the compiler program usually saves the new machine code version of the program on disk, and the original human-readable text is no longer needed (although it is good practice to save it as well).

One Line at a Time

The second method of translating a program text into machine code is to *interpret* it. Remember that the compiler translates the whole program all at once. An interpreter program, on the other hand, takes each line of text, converts it into machine code, runs it, and then goes to the next line.

Compilers and interpreters each have their advantages and disadvantages. Compilers tend to produce

Q and A continued

programs that run very fast, but the programs often take up a lot of memory. Also, the process of compiling a program can sometimes take many minutes. Interpreters are usually slower in operation, but they are much handier for interactive program development. The programmer doesn't have to wait for a program to compile.

The Best of Both

Systems programmers have tried to combine the best features of compilation and interpretation. One way is to make up a very small and fast interpreter (sometimes called a "virtual machine" because it simulates a computer that doesn't exist) and then compile the source text to run on the interpreter. This means that a computer program can be stored in a very compact form and be run at a higher speed than one would expect from an interpreter.

Most versions of Logo are interpreted, but some clever measures have been employed to speed them up and reduce memory requirements. One of these is called "tokenization." It is similar to the idea of a combined compiler / interpreter. In some versions of Logo, you may have noticed that a program runs a bit (oops, sorry!) more slowly the first time than it does subsequently. This is because there is some streamlining of the code going on during its first run. Newer versions of Logo do the streamlining before the program is ever run.

As of this writing, ExperLogo for the Macintosh is the only version of Logo for microcomputers that is compiled. If you have ever seen a demonstration of Exper-Logo, you know that Logo can be *fast!* Coral Software's Object Logo (due out some time next summer) will also be a true compiler.

David Thornburg has often pointed out that it is important that compiled versions of Logo be developed if Logo is to be taken seriously as a computing language. On the other hand, the interpreted versions that are commonly available seem to be entirely appropriate for many classroom uses.

I hope this information helps clear up some of the mystery of the words *compiled* and *interpreted.*

• • • • •

Each month, I will answer selected questions in this column about Logo programming techniques and the thinking which leads to programming. If you have a question, send it to: Jim McCauley, 876 East 12th Avenue, #4, Eugene, OR 97401. Please enclose a stamped self-addressed envelope. You may also send questions to me on CompuServe at 70014,1136.

Jim McCauley is a graduate student at the University of Oregon, studying with David Moursund, and has written Logo articles for many national publications.

NLXual *Challenges*
by Robs Muir

Logo Collectibles

From this side of the geographic fence, I'm sorry to report that Logo is in jeopardy in California's schools. This is a growing and serious problem. (I hesitate to bring up our problems since Los Angeles' high temperature today is 82° and Milwaukee's high is 1°!)

California's problem is *not* the weather, however; California's problem is that Logo is about to become institutionalized in elementary schools. On the surface, that sounds great, right? Yet, Logo won't be institutionalized in the "right" way. Instead, we'll do to Logo what we did to New Math. The buzzwords out here are *"integrate* computers into the curriculum;" in other words, "...don't change, subsume!"

Logo's principal attraction (for me, anyway) was its outspoken advocacy of child centered educational environments in which children could take an active role in structuring their own knowledge. Logo's promise for a more humane classroom where children and adults could share in meaningful intellectual discoveries is about to be lost in our state's drive for a more "productive" educational industry. "Back to Basics" is a killer; it will surely kill the enthusiasm of our students for this new technology if we allow educational policymakers to dictate continuums, scope and sequence charts, and performance objectives by grade level without some sensitivity to the individual's learning needs.

Sound familiar?

As a teacher, I need to justify to myself, my parents, my students, and my administrators that my classroom - with its noise, excitement, and Logo - is as effective a place as that stuff of principals' dreams: the quiet room where every child is on the same page and same paragraph. Logo teachers need to demonstrate that we *can* teach the content of the State Mathematics Framework while playing turtles.

Let's keep this question (challenge?) in mind as you try to solve some of these oldies but goodies.

Focused Challenge # 1

Should my kids memorize the value for pi (π) from a textbook or solve a Logo challenge instead? Draw a square and inscribe a circle within it. (See NLXual Challenges, January 1984.) Can you solve this problem without resorting to π? Does this problem require you to "discover" and learn about π?
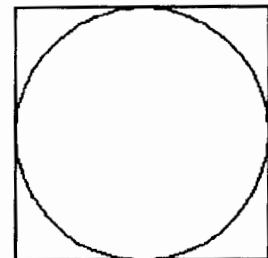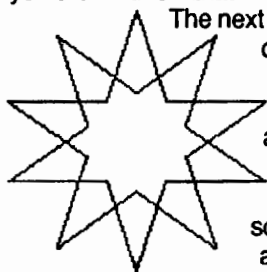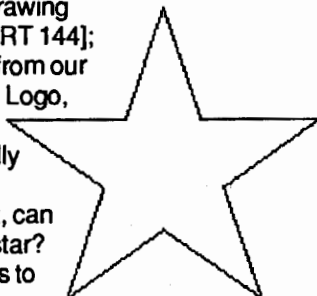
NLXual Challenges continued
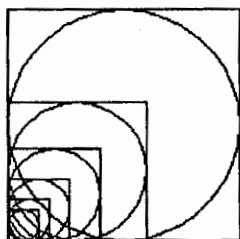
Generalize your solution so that you can draw a square of any size and the corresponding circle; you'll probably learn π more concretely than you did in your junior high school geometry class.

Focused Challenge # 2

We all know the star drawing algorithm, REPEAT 5 [FD 50 RT 144]; that "lesson" was given to us from our teacher training workshops in Logo, or books that we used to learn Logo. Did any of us really discover the Total Turtle Trip Theorem for ourselves? First, can you create this variation of a star? The next task is to create a procedure (or two) with an input so you can draw this double star in any size. (See NLXual Challenges, December 1983.) I have yet to see this done without resorting to some trigonometric functions, such as sine and tangent. Can it be done with "pure" geometry, does anyone know?

Focused Challenge # 3

Ella Pfeiffer's POLYCIRC problem (See NLXual Challenges, November 1985.) remains a superb example of a challenge for high school (?) mathematics students which forces them to confront generalizations about simple polygons and circles. Draw a circle inscribed within an n - sided polygon. Here is a triangle, a pentagon, and a septagon.
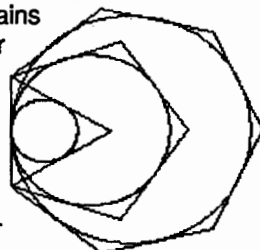
Become Famous

The Logo community needs a collection of such simple focused problems which clearly address fundamental concepts. This could become a way of getting Logo legitimized by the establishment, since we can demonstrate how Logo can "teach" basic concepts. Notice that we can preserve the notion of discovery and invention through "problem solving," another hot buzzword.

Let's catalog some simple challenges that focus on specific pieces of knowledge, such as π and the basic trigonometric functions shown in the foregoing examples.

What Logo activities or projects have you done, with or without your students, that have successfully communicated a key mathematics concept or two? Send an outline of such an activity and we will create a compendium of Logo problems for others to "grab from." The key here is to isolate a specific concept that will lead a user to "discover" a new idea.

These ideas could be used by more traditional teachers to introduce or reinforce concepts, or they could become the core of a more comprehensive Logo-based math curriculum rich in problem solving.

What a gift this might be to the growing body of Logo literature! With a large community of Logophiles, we might create a massive collection of new activities. This could become the print equivalent of a samba school for Logo teachers.
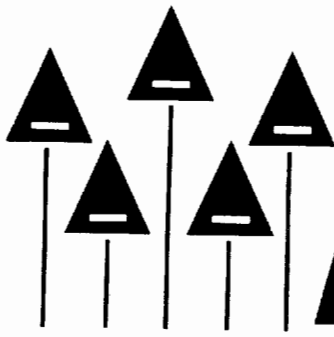
Send your contribution to:

NLXual Challenges
Attn: Robs Muir
1688 Denver Avenue
Claremont, Ca 91711

Be sure to include a stamped self-addressed envelope if you would like a copy of the complete collection of challenges.

Robs Muir is a teacher in the Claremont CA Unified School District and Claremont Graduate School.

Editor..Tom Lough

NLXtra

A Supplement to the National Logo Exchange Newsletter

Imagewriter Overprinting Solved for Apple Logo II !

John Olive, director of the Atlanta-Emory Logo Project of Emory University, Atlanta, GA, has solved the problem of the overprinting with Apple Logo II and the Imagewriter printer! He has written a set of procedures which not only prints out Apple Logo II files but also allows the user to set the left margin and to specify the number of printed lines per page.

He reports, "I had been frustrated with the over-print problem for more than a year when I saw the solution offered for [Apple] Logo (October NLX). I tried all possible ways to send the suggested codes [in Apple Logo II] but was unsuccessful. It was only after looking at the PLOTSTUFF procedures (which drive the Apple Color Plotter) that I was able to figure out the correct sequence of commands for sending control codes to the [Imagewriter] printer via Apple Logo II. After solving the interface problem it was just a matter of deciphering the Imagewriter User's Manual to come up with the appropriate control codes."

Here are John's procedures for Apple Logo II.

```
TO SETPRINTER :LM :PL
OPEN 1
SETWRITE 1
ADDLF
SETLM :LM
SETPL :PL
SETWRITE [ ]
CLOSEALL
END


TO ADDLF
TYPE CHAR 27
TYPE CHAR 68
TYPE CHAR 32
PR CHAR 0
END


TO SETLM :N
IF :N < 100 [ MAKE "N WORD "0 :N ]
IF :N < 10 [ MAKE "N WORD "0 :N ]
```

```
TYPE CHAR 27
PR WORD "L :N
END


TO SETPL :N
( TYPE CHAR 29 CHAR 65 CHAR 64 )
REPEAT ( :N - 2 ) [ ( TYPE CHAR 64 CHAR 64 ) ]
( TYPE CHAR 67 CHAR 64 )
REPEAT ( 66 - :N ) [ ( TYPE CHAR 64 CHAR 64 ) ]
( TYPE CHAR 65 CHAR 64 )
PR CHAR 30
END
```

These procedures are written for the standard 80 column Imagewriter printer with a Super Serial card in slot 1, and assume continuous feed eleven inch paper. They produce hardcopy output at 66 lines per page and 6 lines per inch.

Here is a procedure as an example of how to use SETPRINTER.

```
TO PRFILE :FILE
SETPRINTER 8 58
DRIBBLE 1
POFILE :FILE
NODRIBBLE
END
```

The file :FILE is printed out with a left margin of 8 characters and 58 printed lines per page. Once PRFILE is run, the printer keeps the same configuration for other text dumps using DRIBBLE 1 until SETPRINTER is used with different inputs or the printer is turned off.

Our congratulations and thanks to John for sharing these procedures. They will make life easier for Logo teachers everywhere. But John suggests an additional lesson from the experience. "Looking at procedures that solve similar problems is an important programming strategy which we should encourage in our classrooms." We add our hearty "Amen!!"

NLX readers wishing to contact John may write to Dr. John Olive, Project Director, Atlanta - Emory Logo Project, Emory University, Division of Educational Studies, 201 Fishburne Building, Atlanta, GA 30322.

Other Logo Printing Notices

NLX reader Eadie Adamson uses Applewriter IIe for typing in long programs for Apple Logo II. She then saves the file with .LOGO on the end of the file name, and calls it up from Apple Logo II. She credits Joe Delibero of Packer - Collegiate in Brooklyn, NY, with this idea. Eadie also uses Applewriter IIe to print out Apple Logo II files on the Imagewriter without overprinting. For more information, write Eadie Adamson, The Allen - Stevenson School, 132 East 78th Street, New York, NY 10021.

Have you been frustrated by the inability of Apple Logo to print pictures? Sharon Burrowes' article, "How Can I Print My Apple Logo Pictures?" in the December 1985 / January 1986 issue of *The Computing Teacher* (1787 Agate Street, Eugene, OR 97403) contains a number of useful techniques. Sharon also offers a disk containing programs to print Apple Logo pictures. For more information, write to Sharon Burrowes, 807 College Avenue, Wooster, OH 44691.

Nicole Michaud of Logo Computer Systems Inc., and Eric Nagler of Santa Clara, CA, sent in a procedure for printing Apple Logo II pictures with a Grappler+ card and a parallel printer. Nicole also sent instructions for printing Apple Logo II pictures with a Prowriter II printer. For a copy of either set of instructions, send a self addressed envelope to NLX (Grappler+ or Prowriter), PO Box 5341, Charlottesville, Va 22905-0341.

Terrapin Logo users who want the TEXTEDIT utility program to establish the top and bottom margins of printed pages will now be able to do so. NLX reader Lawrence Alexander has written a set of procedures for serial printers which can be added to TEXTEDIT to maintain top and bottom margins as well as to number each printed page. For a free listing of the Terrapin Logo procedures and documentation, send a stamped self-addressed envelope to Prof. Lawrence Alexander, College of Education, Michigan State University, East Lansing, MI 48824-1034.

NLX Logo ABC Book Planned

The NLX ABC Logo procedures to draw the letters of the alphabet in any size have been distributed free to over 3000 educators. Now it's time to compile the results! *The NLX Logo ABC Book* is in the planning stages, and will contain projects created with the NLX ABC procedures. If you and your students have used the ABC's in any way to create designs, drawings, or other applications, why not send in your work for inclusion in this book? Editor Michael Friendly reports, "I'm excited about the many different projects possible for the book ... drawings made with just one letter, drawings made with words, the list is endless!" Please send procedure listings for your ABC projects to Michael Friendly, 116 Wells Street, Toronto, Ontario M5R 1P3. Include a graphics printout if possible. If you have not yet received your copy of the NLX ABC's, send a stamped self-addressed envelope to NLX ABC's, PO Box 5341, Charlottesville, VA 22905.

Logo Notes

Logo versions are now available in a growing number of different national languages. For information on a particular language, write to Nicole Michaud, Logo Computer Systems, Inc., 9960 Cote de Lisse Road, Lachine, Quebec, H8T 1A1 Canada, or call (514) 631-7081.

Does anyone have any ideas for Logo applications in shorthand classes? If so, send your suggestions to Wanda Printis, 216 Lowden Hunt Drive, Hampton, VA 23666.

The Educational Computing Organization of Ontario (ECOO) announces its Logo conference, "Look to the Learner," to be held March 22, 1986, at York University in Toronto,Ontario. The program features such speakers as E. Paul Goldenberg, William Higginson, and Michael Friendly. This is a full day of Logo activity you will remember for a long time! For more information, write to Pat Davidson, Sloane Public School, 110 Sloane Avenue, Toronto, Ontario M4A 2B1.

The COMPSCI '86 conference will be held on May 12 - 13, 1986 in Clayton, Australia. Also, "Computers in Education: On the Crest of a Wave?" is the 4th Australian Computers in Education conference and is scheduled for August 24 -27, 1986. For more information on these conferences, write to the Computer Education Group of Victoria, PO Box 88, Balaclava, Victoria 3183, Australia.

Biller & Snyder announce the development of a powerful Logo environment to work with Macintosh Logo. For more information, write to Loren Abdulezer, Director, Management Services, Biller & Snyder, 75 Maiden Lane, New York, NY 10038.

Coral Software Corporation has announced the development of Object Logo, an advanced Logo version for the Macintosh. For more information, write to Coral Software Corp., PO Box 307, Cambridge, MA 02142, or call (617) 547-2662.

It's never too early for Christmas! Chuck Thompson has written a set of procedures which draws a beautiful set of tessellated Christmas trees in variable sizes. For a free listing, send a stamped self-addressed envelope to Chuck Thompson, Christmas Tree Tessellation, School of Education, University of Louisville, Louisville, KY 40292.