Logo Trail of Tears
by C. M. Chmielowski

The little boy's sobbing devastated me. He and 23 other second graders had ridden the bus for half an hour to reach the school district's computer lab. And Logo had frustrated him.

The children and their teacher had only two or three hours in my Apple lab. Their brief introduction to Logo must convince them of Logo's value and must provide the tools they need to continue working with the language when they return to their classroom.

In that short amount of time, I must:
1. introduce Logo to all the students and their teacher.
2. work with more than one learner at a time.
3. insure productive and challenging explorations with Logo, not frustrating ones.

Time constraints dictate that the children learn only Logo primitives here in the lab instead of using specially written "Instant" programs. Otherwise, teachers and young children returning to the classroom are frequently confused about how to use the Logo language available at their school.

NO MORE TEARS

Here is a program I developed to help teach the turtle graphic primitives in a way that is not frustrating to young children. It is challenging for first and second graders - more so than an "Instant" program. However, it does not require typing numbers, so its demands are within a 5 or 7 year old's grasp. Children learn to spell the command, and realize they must press <return> after each command. This program can serve as a transition between a kindergartner's use of an "Instant" program and regular Logo. (Note: The space between [D] and the final parenthesis in Terrapin Logo is critical.)

TO HELP
MAKE "ANS REQUEST
PRINT [TYPE FD, BK, RT, LT, OR D]
HELP
END

TO CHECK :ANS
IF :ANS = [FD] FD 10
IF :ANS = [BK] BK 10
IF :ANS = [RT] RT 30
IF :ANS = [LT] LT 30
IF :ANS = [D] DRAW
END

HERE COME THE NUMBERS

When students are comfortable with these turtle commands, they are ready to include numbers with them. The following program permits this, with a turtle that moves slowly enough that children can follow its movements easily regardless of the input numbers.

Its inspiration was Tom Lough's "Slow Turtle" in the September 1983 NLX. The inclusion of a check for numbers larger than 100 prevents accidentally typing a large number. It also discourages endless "wrapping around," thus encouraging more productive experimentation with Logo. When a student is ready to use regular Logo, a simple Ctrl-G is all that is needed. (Note: The space between "LT" and the right parenthesis is critical.)

TO HELP1
MAKE "ANS REQUEST
IF :ANS = [DRAW] DRAW
IF :ANS = [] MAKE "ANS [[]]
PRINT [TYPE A COMMAND, A SPACE, AND A NUMBER.]
HELP1
END
From the Editor

Fighting the Wheel

A friend once told me of her first experience with an automatic drive through car wash. She edged her car up to the entrance, rolled up her windows, and held on to the steering wheel with a death grip as the car was pulled into the path of the robot brushes, sprayers, and polishers. She was terrified. She fought the towing mechanism as it moved her car from one operation to the other. The attendants kept tapping on the car window and shouting, "Let go of the wheel, lady!" But she refused to trust the giant washing machine. After emerging from the other end, she realized that her car was still in one piece, and so was she. Perhaps the machine worked OK.

A couple of weeks later, she had to get her car washed again. This time, she was a little more relaxed, and didn't fight the towing mechanism quite so much.

Lots of things affect us like that. Who can ever forget the anxiety of a first date, a first solo drive in a car, a first day at the first teaching job? What about when we first begin to teach with Logo?

How well I remember my first Logo attempts. One day, I would be too rigid and structured, "holding on to that steering wheel." The next day, I would be too loose. But, gradually, I found ways to teach with Logo which seemed to be effective. I began to relax and enjoy the experience with my students.

Steve Tipps often reminds anxious Logo teachers that Logo learning is developmental; it takes time. We all have to go "through the wash" a time or two before we feel relaxed and confident in what we are doing.

But, do not get the impression that, after taking a four-hour workshop or a three-hour course and teaching with Logo for a year, you can put your feet up. Logo learning has a special way of happening. It arrives quietly, in bits and pieces, as you try different things with your students. But it takes that effort on your part.

If you are having difficulty "letting go" as you begin your use of Logo, you have lots of company. Just don't be too hard on yourself. Time is on your side.

FD 100!
Tipps for Teachers
by Steve Tipps

Number High, Number Low

Learning about equalities and inequalities is sometimes difficult for children. The TEST command with numbers provides another way to explore with Logo in mathematics class. By working with TEST and programming different consequences, children use the computer to test hypotheses about numbers. A number guessing game is one project which can be expanded in many ways.

QUEST FOR TRUTH

The computer obeys consequences, but setting up the TEST and the results is a task for the problem solver. Begin in the immediate mode with a simple TEST. (For this column, I am using Terrapin Logo. In most cases, the conversion to LCSI Logo is straightforward.)

TEST 9 = 9

Many IFTRUE consequences are possible

IFTRUE PRINT [ 9 = 9 ]
9 = 9
IFTRUE PRINT [ TRUE ]
TRUE
IFTRUE REPEAT 9 [ PC 3 RT 180/9 PU FD 50 PD FD 50 PU BK 90 ]

as are many IFFALSE consequences.

IFFALSE PRINT [ FALSE ]
IFFALSE [ CRAZY.STAR ]

In the case of TEST 9 = 9, the result of the TEST is true and only the IFTRUE consequences happen. Nothing at all happens when the IFFALSE consequences are typed. The TEST sets a signal of either TRUE or FALSE. That signal, or flag, stays set up until a new TEST is made.

Students can try writing TESTs and consequences in the immediate mode for any numbers and talk about the expected results.

TEST 100 > 30
"That's true."

IFFALSE PRINT CHAR 7
"Nothing happens."

If the TEST is changed to TEST 100 < 30, the IFFALSE consequences are executed. PRINT [ 100 > 30 ] is unconditional and always executes. Rather than writing in the immediate mode, it is much handier to write a procedure containing consequences.

TO IS.IT.TRUE?
IFTRUE PRINT CHAR 7
IFTRUE PRINT [ YES, IT IS TRUE ]
IFTRUE BG 3
IFTRUE PC 4 SQUARE
...
END

TEST 40 = 40
IS.IT.TRUE?
YES, IT IS TRUE.
...

IS.IT.TRUE? and a companion procedure IS.IT.FALSE? can be expanded with more and more consequences.

NUMBERS GREAT AND SMALL

After testing numbers many times and in many conditions, students should see that three situations are possible when comparing two numbers. They can be equal or unequal. If they are unequal, one or the other will have the larger value.

COMPARE might be a good name for a procedure which will compare any two numbers. Prepare for COMPARE by talking about all the possibilities. A description of COMPARE might be:

Find out if the first number is equal to the second.
If they are equal, print an equality statement and stop.
If they are not equal, find out if the first is larger.
If the first is larger, print that it is.
If the first is not larger, print that the second is larger.
An English description in logical steps is sometimes called pseudocode. Some people use flowcharts to do the same kind of planning. The actual procedure does exactly what is described in the pseudocode.

```
TO COMPARE :FIRST :SECOND
NODRAW
TEST :FIRST = :SECOND
IFTRUE PRINT1 :FIRST PRINT1 [' IS EQUAL TO ']
PRINT :SECOND
IFTRUE STOP
TEST :FIRST > :SECOND
IFTRUE PRINT1 :FIRST PRINT1 [' IS GREATER THAN ']
PRINT :SECOND
IFFALSE PRINT1 :SECOND PRINT1 [' IS GREATER THAN ']
END
```

The important statements of relationships between the numbers are in the procedure, but no one gives this procedure any prizes for excitement. A number guessing game might be a more stimulating way to play with the inequalities.

**GETTING CLOSER**

A simple guessing game would set the number and let you try to guess it.

```
TO TRY :NUMBER
TEST :NUMBER = 14
IFTRUE PRINT [THAT IS IT.]
IFFALSE PRINT [THAT IS NOT IT.]
END
```

Even if the range were restricted to numbers from 1 to 50, the process of guessing would be fairly laborious.

```
TRY 30
THAT IS NOT IT.
TRY 15
THAT IS NOT IT.
TRY 9
THAT IS NOT IT.
```

Forty-six tries later you might guess the correct number. In exasperation, you might even be tempted to peek at the number by looking in TRY.

It would be helpful if students find out immediately that 9 is too low and 30 is too high. Also, 15 is closer than 30. This game gives you no information about the different numbers. Using ideas from COMPARE makes the game much better by providing informative feedback about which number is too high or too low.

```
TO GUESS :NUMBER
TEST :NUMBER = 77
IFTRUE PRINT [THAT IS RIGHT.]
TO GUESS :NUMBER
TEST :NUMBER < 77
IFTRUE PRINT [THAT IS TOO LOW.]
IFFALSE PRINT [THAT IS TOO HIGH.]
END
```

Now the guesser gets information about the guess.

```
GUESS 89
THAT IS TOO HIGH.
GUESS 31
THAT IS TOO LOW.
GUESS 77
THAT IS RIGHT.
77 IS THE NUMBER.
```

Information about the guess makes the game more interesting, but not exciting. Programming other consequences such as coloring the background or drawing pictures would make the game more entertaining.

A guessing game might proceed through six or eight stages as students refine and add to it. The teacher serves as a programming consultant in the project.

**GUESS** is only a skeleton program for students and teachers to improve into an educational game on numbers. Starter programs like this should stimulate problems because they don't do everything you want them to do.

**FIXING IT**

One thing that this program doesn't do is give you an easy way to change the mystery number. It is possible to change the number by editing GUESS and putting in a new mystery number. But you must insert the new number in three separate places. You want an even easier method. Another way to change the number is to assign a value to IT with a MAKE statement.

```
MAKE "IT 35
```

Now, GUESS can be changed to GUESS.IT, to use the value of assigned to the variable IT. Of course, the variable IT must appear in three separate places in the procedure.

```
TO GUESS.IT :NUMBER
TEST :NUMBER = :IT
IFTRUE PRINT [THAT IS RIGHT.]
TO GUESS.IT :NUMBER
TEST :NUMBER > :IT
IFTRUE PRINT [THAT IS TOO HIGH.]
TO GUESS.IT :NUMBER
TEST :NUMBER < :IT
IFTRUE PRINT [THAT IS TOO LOW.]
END
```

Now, GUESS can be changed to GUESS.IT, to use the value of assigned to the variable IT. Of course, the variable IT must appear in three separate places in the procedure.
Tipps continued

IFTRUE STOP
TEST :NUMBER < :IT
IFTRUE PRINT [THAT IS TOO LOW.]
IFFALSE PRINT [THAT IS TOO HIGH.]
END

What if you want to play GUESS.IT solitaire? The excitement pales if you have to MAKE it and GUESS.IT, too. Logo can randomly choose its own mystery number.

TO SET.IT
RANDOMIZE
MAKE "IT RANDOM 100
PRINT [TRY TO GUESS THE MYSTERY NUMBER.]
PRINT [IT IS A NUMBER FROM 0 THROUGH 99.]
END

SET.IT
TRY TO GUESS THE MYSTERY NUMBER.
IT IS A NUMBER FROM 0 THROUGH 99.

SET.IT sets a new number for IT, and the guessing game can start again. RANDOMIZE is used to reset the random number list.

NUMBERS UP

COMPARE, TEST, and GUESS.IT might also be used for an estimation game. Announce a target number, and several arithmetic problems. A simple procedure with PRINT might display this information.

THE TARGET IS 357

WHICH PROBLEM GIVES THE ANSWER CLOSEST TO 357?
42 x 8
21 x 18
52 x 7
11 x 33

WHICH ANSWERS ARE TOO HIGH?
WHICH ANSWERS ARE TOO LOW?
ESTIMATE, THEN USE LOGO TO CHECK.

Logo provides many ways to work on this problem - from using GUESS.IT to calculating in the immediate mode.

MAKE "IT 357
GUESS.IT 21 * 18
THAT IS TOO HIGH.
COMPARE 357 52*7
364 IS GREATER THAN 357

PRINT 11*33
363
PRINT 11*33 - :IT
6

Estimation skills can reveal much about students' understanding of arithmetic. Logo is a tool for testing estimates and increasing understanding of numbers. Whether the number is too high, too low, or just right, Logo helps everyone win by learning about numbers.

Next month, Numbers in Between.

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX. He presents Logo workshops for school systems throughout the United States.

NLXionary

A Lectionary of Selected Logo Readings

by Griff Wigley

This month's selections are from panelist papers written for Logo 84, the National Logo Conference held at MIT this past June.

"Creating Logo Cultures," by Dan Watt
-Watt's contention is that Logo needs a supportive "culture" to be successful in the way that reading and writing have been. By culture, he means an environment in which people regularly engage in activities related to the skill in a multitude of ways. He thinks that expecting teachers to create successful Logo experiences for students after only a few days of learning Logo is analogous to expecting teachers to teach reading and writing successfully after only a few days of learning to read and write.

-However, he cites several examples in which Logo utilization has gone beyond the mechanics of learning the language, showing that "Logo can indeed serve as a catalyst and an organizing principle for the kind of human culture we are all trying to create ... rich with experiences, ideas and people that break down the barriers between disciplines, between teacher and learner, and between people with different learning styles."
NLXionary continued

Watt doesn't mention that this kind of environment can be created without one smidgeon of Logo, which I find slightly disturbing. His reading / writing analogy and closing statements seem to elevate Logo's importance needlessly. I prefer to differentiate Logo from the philosophy behind it, seeing it instead as one of many possible threads in the current fabric of effective learning environments.

"About That Rose Garden: Remarks on Logo, Learning, Children and Schools," by William Higginson

Philosophy delivered with delightful wit by the editor of Logophile. Higginson lampoons research that attempts to measure Logo's impact in behavioralistic and quantitative terms. He reminds us that Logo is a current battleground "for an old philosophical dispute," and cites writers in education and in other disciplines whose philosophy is parallel to that of Logo. Rather than simply asking, "What do children learn from Logo?" he prefers to ask, "Is it the case that Logo can be used to aid the intellectual and social growth of children in the manner described by Papert?" The man has data to answer the question.

"So What Do Children Learn with Logo?" by Rina Cohen

A dialogue between a Logo educator (the author) and a Logo mathematician illustrating issues involved in evaluation, some typical negative outcomes involving the implementation of Logo, and some possible solutions. One interesting idea for evaluating Logo-related learner outcomes, even with a group of students who are learning Logo from a teacher only recently afflicted with "Logo fever" is to measure specific cognitive skills such as map reading and directionality. Trying to measure Logo's potential for fostering a love of learning, a sense of mastery and power, cooperation, and all the other components that contribute to a student developing his or her full ability is too risky because it depends too much on the individual teacher's personal characteristics and philosophy of education.

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.

Logo 84 Proceedings Available

NLX readers may obtain a copy of the Logo 84 conference proceedings which contain the papers mentioned in this month's NLXionary, in addition to many other papers and abstracts. Also included is an extensive Logo bibliography containing listings of books, periodicals, and articles. To obtain your copy, send a check for $15 US to MIT Logo Group/Publications, NE43-438, 545 Technology Square, Cambridge, MA 02139.

Turtle Tips

by Donna Lanyi and Jane Toth

What takes the most time in writing a procedure? Typing it in!! By now, you have noticed that many students spend a great deal of time "hunting and pecking." Keyboarding skills do play a major role in developing computer skills.

Teacher Hints

A useful teaching aid that we have developed is a posterboard size picture of the keyboard. We have color coded the home row keys to match a set of color coded hands that accompany the picture. We also supply students with a desktop laminated keyboard. This allows students to practice keyboarding skills without using computer time.

We use keyboarding practice books for children that start with the home row keys and build. Sometimes we use old typing books for exercises. Our students also practice typing procedures so that they familiarize themselves with the most commonly used Logo keys.

If you can find an old typewriter, place it in a corner of your room and allow students to use it for practice. Perhaps you could develop a learning center for these activities.

Commercially produced practice keyboards are available for most popular computers from Computer Practice Keyboard Company, 616 9th Street, Union City, NJ 07087, for $4.95 each. The price drops to $3.00 if you order 12 or more. Be sure to specify your computer make and model.
TurtleTips continued

PROCEDURE OF THE MONTH

For November, have your students design a picture to depict Thanksgiving. Ask your students to match our Indian face, or create one of their own. If you are interested in a free listing of this procedure, please send us a self addressed stamped envelope.

---

We invite elementary teachers and students to send procedures or ideas to be shared. Donna Lanyi and Jane Toth, Beall Avenue School, 716 Beall Avenue, Wooster OH 44691.

---

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

---

Listful Thinking

by

Glen Bull and Paula Cochran

Count Your Blessings

For thousands of years, people in cultures around the world have celebrated the time of harvest. In America, Governor Bradford was the first to declare a day of Thanksgiving among the colonists of Plymouth in November, 1623. Everywhere, such a celebration causes people to reflect on their good fortune, families, favorite foods, (football, perhaps), -- and the list goes on. Which brings us to Logo (subtle, huh?).

This month, we would like to help you explore how to COUNT items in a list and ADD items to a list. We'll show you how to INSPECT your lists, and use them to make Thanksgiving story starters.

WHAT'S IN A LIST?

As you recall from last month, Logo can be used to MAKE a list as follows:

MAKE "FEAST [TURKEY CORN YAMS]
MAKE "PILGRIM [PRISCILLA MILES]

You can ask Logo to print out the names of your lists and their contents with the command PONS (Print Out Names). (For MIT Logo, use PO NAMES.)

PONS "FEAST IS [TURKEY CORN YAMS]
"PILGRIM IS [PRISCILLA MILES]

As we saw last month, we can also PRINT the contents of a list.

PRINT :PILGRIM
PRISCILLA MILES

AT LAST COUNT

Let's suppose we want to know how many items are in a list without having to count them ourselves. Logo can do this and then tell us the number, using the COUNT primitive. (If you are using Version 1 of Terrapin or Krell Logo, COUNT must be defined as a procedure such as the one at the end of this column.)
Listful Thinking continued

COUNT outputs a number. We must tell Logo what to do with that number. For example, we could PRINT it:

```
PRINT COUNT :FEAST
3
PRINT COUNT :PILGRIM
2
PRINT (COUNT :FEAST)+(COUNT :PILGRIM)
5
```

MORE PILGRIMS AND MORE TOOLS

In previous columns, we have been encouraging you to use the procedure PICK as a tool. ADD and INSPECT are two more procedures which can be used in this way. Your students can use them much as they would use REPEAT or other primitives. ADD can add items to an existing list with the LPUT command.

```
TO ADD :ITEM :LIST
MAKE :LIST LPUT :ITEM THING :LIST
END
```

This procedure can add the name SUE to the list of PILGRIMs.

```
ADD "SUE" PILGRIM
```

To check,

```
PRINT :PILGRIM
PRISCILLA MILES SUE
```

INSPECT prints the contents of a list in a neat, one-per-line format. (See a slightly different procedure at the end of the column for Terrapin and Krell.)

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

```
INSPECT :PILGRIM
PRISCILLA MILES SUE
```

Here's a procedure that makes it easy to add to the list of pilgrims and count them, too. The command READLIST waits until you type a name on the keyboard. Then ADD accepts the name and adds it to the list of PILGRIMs. (If you are using Terrapin or Krell, substitute REQUEST for of READLIST in the procedure.)

```
TO ADD.PILGRIM
PRINT "NAME A NEW PILGRIM"
ADD READLIST "PILGRIM"
PRINT []
COUNT.PILGRIMS
END
```

```
TO COUNT.PILGRIMS
PRINT "HERE ARE THE PILGRIMS:
INSPECT :PILGRIM
PRINT []
PRINT (SENTENCE [THERE ARE] COUNT :PILGRIM [PILGRIMS NOW!])
END
```

Here's what happens when ADD.PILGRIM is used.

```
ADD.PILGRIM
NAME A NEW PILGRIM
TOM
```

```
HERE ARE THE PILGRIMS:
PRISCILLA MILES SUE TOM
```

```
THERE ARE 4 PILGRIMS NOW!
```

You could make a similar procedure to add to the list of foods.

```
TO ADD.TO.FEAST
... ...
END
```

You may have noticed that one or two of the lines in the ADD.TO.FEAST procedure have been omitted. In a Logo-like fashion, we invite you to consult ADD.PILGRIM and develop your own personal ADD.TO.FEAST procedure. If you are emphasizing nutrition, your class could establish a separate list for each food group, for example.

ALL THE FIXINGS

It's easy to make a mistake typing a name. Suppose that, instead of TOM, you actually wanted TIM in the list of pilgrim names. In Logo, you can edit lists of names, just as you can edit procedures. Just type EDNS; (With Terrapin and Krell, type EDIT NAMES.)

When you edit names, you go to the editor. Instead of a procedure, you see something that looks like this:

```
MAKE "FEAST [TURKEY CORN YAMS]
MAKE "PILGRIM [PRISCILLA MILES SUE [TOM]]
```
Listful Thinking continued

Now use the edit keys to move over to TOM and fix the spelling, just as though you were editing a procedure. Then, exit the editor in the usual way to define the edited lists.

(By the way, [TOM] appears in the pilgrim list instead of just TOM. This is because ADD.PILGRIM is using READLIST to provide the input to ADD. This allows those with names such as MARY JANE to be ADDed to the pilgrim list.)

THE FIRST THANKSGIVING

After the children in your class add their names to the list of pilgrims, they can be in a story. As we saw last month, the PICK procedure is used to pick an item randomly from a list. (See listing at the end of this column.)

PRINT PICK :PILGRIM MILES

In a sentence, PICK works like this:

PRINT (SENTENCE PICK :PILGRIM {was in charge of the feast.})

TIM was in charge of the feast.

What can you do with a list of pilgrims and a feast menu? After your children have developed the lists, they can use a short procedure (such as the one which follows) to generate individual story starters. The story starters can be used as a basis for oral story-telling, a writing assignment, an art project, etc.

TO STORY
PRINT (SE [THERE WERE] COUNT :PILGRIM [PILGRIMS IN THE COLONY.])
PRINT (SE PICK :PILGRIM [was in charge of the feast.])
END

A story starter might look like this.

STORY
THERE WERE 3 PILGRIMS IN THE COLONY.
SHEILA WAS IN CHARGE OF THE FEAST.
EVERYONE BUT DICK LIKED TURKEY.

Each story starter has different names. As more pilgrims are added to the list, the number in the story changes, too.

ADD.PILGRIM
NAME A NEW PILGRIM
KEELAN

HERE ARE THE PILGRIMS:
MILES
DICK
SHEILA
KEELAN

THERE ARE 4 PILGRIMS NOW!

With the new pilgrim added, the story starter might change to something like this:

STORY
THERE WERE 4 PILGRIMS IN THE COLONY.
MILES WAS IN CHARGE OF THE FEAST.
EVERYONE BUT KEELAN LIKED YAMS.

We hope that you and your students enjoy COUNTing and ADDing your blessings this month.

Next month: Making a List and Checking It Twice!

PROCEDURES

Here is last month's PICK procedure:

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

If you are using Version 1 of Terrapin or Krell Logo, the following procedures are needed with PICK.

TO ITEM :NUMBER :LIST
OUTPUT ITEM.LOOP :NUMBER :LIST
END

TO ITEM.LOOP :NUMBER :LIST :TALLY
IF :LIST = [ ] OUTPUT (SE [THERE ARE ONLY] :TALLY [ITEMS IN THIS LIST.])
IF :NUMBER = :TALLY OUTPUT FIRST :LIST
OUTPUT ITEM.LOOP :NUMBER BUTFIRST :LIST :TALLY + 1
END

TO COUNT :LIST
OUTPUT COUNT.LOOP :LIST 0
END

TO COUNT.LOOP :LIST :NUMBER
IF :LIST = [ ] OUTPUT :NUMBER
OUTPUT COUNT.LOOP BUTFIRST :LIST :NUMBER + 1
END
Listful Thinking continued

This is INSPECT for Terrapin and Krell Logo.

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

Glen Bull is a professor in the University of Virginia's School of Education, and teaches Logo courses at both the graduate and undergraduate level. Paula Cochran is a reformed English major who studied linguistics at the University of Cambridge. She is now a speech-language pathologist working with language-disabled children.

Teacher to Teacher

Logo Exploration: An Adventure for Middle School Students reviewed by Jim Fry

"You are about to begin an adventure. Imagine that you are sailing to a chain of islands in the Pacific Ocean. The purpose of the trip is to explore the islands and learn all you can about what is there: mountains, rivers, volcanoes, birds, other animals, flowers and trees."

Thus begins Explorer's Guide to Apple Logo. The authors state that their book is a guide to Logo, like a map which shows where places are located and gives some information about them. But, as with a map, you still need to visit the places yourself and make your own discoveries. And, as with a guide book, you might not want to proceed in the order of the pages; you might want to skip around.

WHAT A BEGINNING

What an exciting way for a teacher to introduce Logo! After I read the first chapter, I was brainstorming with myself about all the ways I could tie in the explorer idea with activities for my students to use both in the classroom and at home.

The book itself is a six chapter workbook, with an appendix of eight sections dealing with such things as a summary of Logo commands, glossary, information on handling disks, using a printer, and other information for teachers.

The six chapters are:
1. Introduction - discusses the idea of exploration and sets the stage
2. Activities without the Computer - an important chapter for precomputer work, especially for classes with limited computer time
3. Commanding the Turtle - suggests beginning activities for turtle graphics
4. Procedures - includes working in the editor and saving procedures
5. Shapes - explores geometric concepts and introduces the idea behind the Total Turtle Trip Theorem
6. Making Designs - expands on Chapter 5 and introduces superprocedures

The book is organized in a helpful manner. Reference pages have colored borders to make them easy to spot. Activity pages with ideas and suggestions for things to try make up the bulk of each chapter. The more important activities are marked clearly. Challenger activities for more advanced students are also included. There is plenty of room for writing and sketching on the pages.

SPRINGBOARDS AND STARTING POINTS

Many ideas in the book can become springboards for additional activities. For example, after the students are introduced to the turtle, the concept of degrees for use with the RIGHT and LEFT commands is developed. The authors indicate how map making and navigation employ degrees, how the Air Force uses the numbers on a clock to judge position, and how degrees relate to these numbers. Directions to make a turtle clock are included. There are also additional examples in the back of the book for teachers to use.

This part of the book would be a perfect addition to a unit on compass use. Mapping activities could be developed by teacher and students using the turtle to draw treasure maps or mystery island maps with a compass rose to show direction. Students could then give compass directions and turtle steps (old pirate maps used paces, I'm told) to friends to find the hidden treasure or the mystery location.
Teacher to Teacher continued

FOR WHOM?

Upper elementary or middle school students would be able to use Explorer's Guide to Apple Logo on their own, with guidance from the teacher. The reading level is too difficult for most students below fifth grade. For teachers in lower grades, the book could serve as a source of ideas for simpler student activities.

Explorer's Guide to Apple Logo lives up to its name as a guide book to help students and teachers become explorers in the exciting world of Logo. A teacher's manual and a review set of 15 spirit duplicating masters are also available.

Explorer's Guide to Apple Logo (LCSI Version) by Joan Moscovitch, Peter von Mertens, and Maggie Neville Holmes, $12.95, Hayden Book Company, 10 Mulholland Drive, Hasbrouck Heights, NJ 07604. (201) 393-6000. An edition for MIT Logo is available also.

If you have comments, questions, or concerns about this review, please contact Logo Information For Teachers (LIFT) at P.O. Box 5396, Plymouth, MI 48170.

Jim Fry teaches with Logo at Novi Elementary School in Novi, MI, and is a co-founder of LIFT.

Q and A

by Jim McCauley

Q. I'm really puzzled by OUTPUT and its use in recursive procedures. I was working with a procedure like:

```plaintext
TO ANY.GREATERP :N :LST
IF EMPTYP :LST [OUTPUT "FALSE"
IF (FIRST :LST) > :N [OUTPUT "TRUE"
OUTPUT ANY.GREATERP :N BUTFIRST :LST END
```

which checks to see if there are any numbers greater than N in a list called LST. What took me a long time to figure out is this: Not only must you arrange for the procedure to OUTPUT TRUE and FALSE, but you must also OUTPUT the recursive call to ANY.GREATERP! Why? This seems to me to be one of the features of Logo that is not really well documented.

<[-?-]>

A. Let me open my response by agreeing immediately with your last comment. While there are many examples of this feature of LCSI and MIT Logos in the documentation supplied, it is left to the user to "read between the lines" to pick up on this technique. Indeed, most of the trade books on Logo give it only the lightest treatment. The authors assume that it is "logical" (and it is ...), but I agree with you that it is not intuitively obvious.

Let me try to make it a bit more comprehensible. Take a procedure like:

```plaintext
TO ADD.N :N :LST
IF EMPTYP :LST [OUTPUT [1]]
OUTPUT SENTENCE (FIRST :LST) + :N
ADD.N BUTFIRST :LST END
```

This adds the number N to every number in the list LST and outputs the modified LST when the job is done. Its operation is classically recursive: until LST is empty, it "holds on" to the first number in LST, adding N to it, then creates another "copy" of itself to send the BUTFIRST of LST for further processing.

Each of the copies creates other copies until LST is empty. The last copy receives an empty LST and outputs it.

NLX Notes

It's not too early to start thinking about Christmas gifts for special people. How about a gift subscription to the NLX for a teacher relative or friend? Send $25 and a written message to NLX P.O. Box 5341, Charlottesville, VA 22905. We will mail a special Christmas gift notification which includes your written message, and start the subscription with the January issue.

The next edition of the NLX Logo Research Directory will be published soon! If you are conducting educational research relating to Logo, you should be listed. Please send your name, address, and a summary of your research interests and activities to NLX Research Directory, Attn: Regina Saponas, P.O. Box 5341, Charlottesville, VA 22905.


Q and A continued

Like STOP, OUTPUT halts the execution of a procedure. But, unlike STOP, it returns some information to the procedure that called it into action.

THE CHAIN GANG

Imagine all of these copies of ADD.N as a chain, each copy linked to the one that called it and the one it called. When LST is empty, the chain "bottoms out," and an empty LST is output to the next-to-last copy of ADD.N.

Now that copy OUTPUTs a sentence made of two things: the number to which it added N and the empty LST it received from the last copy of ADD.N.

The LST it outputs is treated by the copy "above" it in a similar way, a process which continues until the "top" copy outputs the reconstructed LST to the primitive or procedure that called ADD.N in the first place. Thus,

SHOW ADD.N 4 [21 14 52]

produces

[25 18 56]

(If you use MIT Logo, change the first line in ADD.N to IF EMPTY? :LST OUTPUT [] or IF :LST = [] OUTPUT [], and test it with ADD.N 4 [21 14 52]. There is no SHOW primitive in MIT Logo.)

I hope this helps you to understand why the last line of ADD.N contains OUTPUT. It is necessary to return the processed LST to all the "higher" copies of ADD.N so that the entire LST can be reconstructed. If you'd like more practice in understanding how this works, Harold Abelson has a nice illustration in Section 8.1 of his Logo primer, Apple Logo (for LCSl), or Logo for the Apple II (for MIT Logo), published by Byte/McGraw Hall.

BACK TO ANY.GREAT ERRP

In your ANY.GREAT ERRP example, things are only slightly different. ANY.GREAT ERRP is not trying to preserve the structure of a list that it will later reassemble; instead, it is simply taking the list apart and looking for a number greater than N. If it finds one, it tells the world by outputting TRUE and dropping the task. If it gets
to the end of LST without finding what it's looking for, it outputs FALSE and halts. So it seems a bit unnecessary to OUTPUT the recursive call to ANY.GREAT ERRP, since we don't expect to reconstruct the list that has been torn apart.

ANY.GREAT ERRP is a "tail recursive" procedure, a special type that resembles a loop. It involves a system programmer's trick. Usually, recursive procedures eat up memory like crazy, since there are all those copies lying about, waiting for their "offspring" to return to them with some output.

LOGO INTERPRETED

Often, Logo designers save memory by making the Logo interpreter smart enough to detect when tail recursion is called for. These smart interpreters simply handle such procedures as though they were loops. In order to get the trick to work, though, the syntax must remain consistent, so OUTPUT must appear in the line that makes the recursive call.

Incidentally, some Logo interpreters are smarter than others with respect to this. I've tested all the Logo implementations from LCSl, and they treat both tail recursive operations (procedures that output) and tail recursive commands (procedures that don't output) as I have described. The MIT versions for the Apple from Krell and Terrapin treat tail recursive commands in this way, but not tail recursive operations.

Usually, this is not a serious limitation, since you'd have to have more than eighty items in a list that was being processed recursively before you ran past the interpreter's capacity to store the previously created copies.

I hope this has helped you understand better all that happens when you use OUTPUT recursively. Good luck!

(Ed. note: If you have questions about Logo programming techniques or the thinking which leads to Logo programming, send them to NLX Q & A, PO Box 5341, Charlottesville, VA 22905. Each month, Jim McCauley answers selected questions in this column.)
by Hillel Weintraub

Some very exciting developments in the growth of Computer Assisted Learning (CAL) have occurred in Japan in the past few months. I attended a fantastic CAL meeting in Osaka, organized by Shunichi Yamazaki. Hands-on sessions with the new Fujitsu Logo were presented to an audience of mostly beginners by Masaru Takahashi. Takito Totsuka and I made presentations also. Takito is a primary school teacher who is using Logo to increase his students' understanding of natural phenomena such as leaf structure and movement of planetary satellites.

AN ENTHUSIASTIC REACTION

These presentations got the audience members very enthusiastic about the real potential of Logo in CAL. Much discussion centered around Logo itself. Is it Logo that is magical? Or, perhaps the magic comes from how it is used by creative teachers to help learning take place.

We issued ourselves this warning: "We must remember that it is not Logo itself that we are advocating, but what we feel Logo represents: a learning environment in which students can discover important things about themselves and the world."

Fujitsu Logo

Here is more information about the new Fujitsu Logo. It runs on the Fujitsu 7 series of computers. The newest one, the FM-77, is quite a machine: two drives, a 14" RGB monitor, and a detached keyboard for about $1000 discounted. A major point of the Fujitsu advertising campaign is the bundling of Logo: that is, it comes with the machine. This is pretty amazing, considering that most people in Japan had never heard of Logo a few months ago!

Fujitsu Logo was created by Logo Computer Systems Inc. (LCSI) and is similar in operation to Apple Logo II. (Ed. note: for a review of Apple Logo II, see NLX Sept 1984.)

The Fujitsu keyboard has been especially designed to make input of common Logo symbols such as quotes and dots very convenient. I believe we will be seeing more of this type of user consideration in the future as Logo is designed for other specific systems.

Some other special features of Fujitsu Logo include: a 3 channel music system; dual 88B09/Z-80 microprocessors for speedy turtle graphics; over 4000 nodes available; an editor for 32x16 sized shapes; 4 turtles; and the unique ability to handle Japanese characters (Kanji) by treating them as turtle shapes!

Of course, Logo is not meant to use Kanji in word processing, but to be able to use them in graphic applications or in translation games is a nice addition.

Fujitsu Logo has an enhanced capability to handle property lists, including several special property list primitives. The use of Logo for keeping records is emphasized.

There is no news about the release of Fujitsu Logo in the USA, but I'm sure that the features on this extraordinary version will affect the design of future Logo systems released everywhere!

If you have any questions about Logo in Japan, write to me at Doshisha International High School, Tatara, Tanabe-cho, Tsuzuki-gun, Kyoto-fu, Japan 610-03.

Next month, I'll bring you up to date on Epson's mighty Logo for the QX-10. Until then, MAE 100!

Hillel Weintraub is a teacher at Doshisha International High School in Kyoto-fu, Japan, and is the president of the Society for Microcomputing in Life and Education (SMILE).

64K for the Apple III

Have you been wanting to run Logo on your Apple III, but could not get it to work, even in the Apple II emulator mode? This because only 48 K can be addressed. There is a new emulator board now available from Titan Technologies which has 64 K of addressable memory for the Apple III. For the first time, this makes it possible to run the "regular" Logos on the III. There is also a configuration available with 128 K. For more information, contact Debbi Kendall, Titan Technologies, Inc., 310 West Ann Street, Ann Arbor, MI 48104, or call (313) 662-8542.
Logo Disserts:
Dissertations Dealing with Logo
by Barbara Elias

Logo learning and geometry concepts naturally go together. Turtle geometry is often used to introduce Logo. Many geometry concepts can be explored in this interactive animated environment. When considering geometry concepts for the young child, however, we must be mindful of how the child tends to view shapes, lines, and angles.

For example, we see a square as a square - rigid and fixed. Young children view lines and shapes from a more topological perspective; that is, they see the shapes as being flexible and able to bend rather than as rigid and fixed.

THE QUESTIONS

Is it possible that a Logo learning experience can influence the way young children think about geometry and perhaps lead to more systematic thought processes? L. P. Rieber's masters thesis examines both this question and a broader one: "Can Logo learning influence the developmental characteristics of young children?" The latter is the kind of question that Papert challenges us to examine.

Now that the microcomputer and Logo are classroom tools, questions like this can be investigated in the regular classroom setting. It was in such a setting that second graders explored the concepts of angle, degree, measurement, and the Total Turtle Trip Theorem. The teacher, aided by fifth grade tutors, used a guided discovery approach during the three month long study. Children were instructed in the basic turtle commands, geometry, and measurement. There were encouraged to "play turtle" as they worked.

THE RESULTS

Pretest and posttest data were collected from Piagetian measures, a teacher designed geometry test, and interviews. There was a significant difference in the performance on children in the experimental group on all measures. Logo was a new experience for the children in this study. It was also an enjoyable experience.

The results suggest that Logo learning can have an impact on the development of systematic thought in the young as well as provide a means of examining developmental characteristics from an additional perspective. More comprehensive studies in this area are needed.


(Ed. note: Each month, Barbara Elias highlights a dissertation or thesis dealing with Logo. She is accepting copies of recent research results for the NLX Dissertation and Thesis Repository. Reports may be mailed to her in care of NLX, PO Box 5341, Charlottesville, VA 22905.)

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 Virginia.

Michigan Logo Users Group Formed

The Oakland County (MI) Logo Users Group functions as a forum where local educators can share teaching styles, materials, and ideas for implementing Logo. Through this group, a network of Logo users and resource people have been identified. The group has started a list of resource materials, including books, magazines with Logo columns or special Logo issues, articles, and manuals.

Future meetings will include an update of local district Logo activities, especially the district curriculum writing projects. These projects focus on either how to integrate Logo into existing literacy classes as a way of introducing students to the Logo language and environment, or how to integrate Logo into the existing curricular areas for use as a tool in subject areas and a means of problem solving in general.

For more information on the Oakland County Logo Users Group, contact Lynn Lieberman, Oakland Schools, 2100 Pontiac Lake Road, Pontiac, MI 48054, or call (313) 858-1896.
Turtle Songs

Meet our friend, the Logo Turtle; say, "How do you do?"

It is fun to tell the Logo Turtle what to do.

1. Forward fifty, turn right ninety, four times makes a square.
2. Forward fifty, right one twenty, three times and it's done.
3. Draw a square, then forward fifty, turn right thirty now.
4. Logo Turtle makes a house by joining different parts.

Logo Turtle, extraordinaire!
Our triangle is a perfect one!
Add triangle and we've got a house! (WOW!)
We can make a world by joining hearts!

Meet our friend, the Logo Turtle; say, "How do you do?"

With our Logo Turtle, there is much more we can do.

Copyright (c) 1984 Tom Lough. All rights reserved, with the exception that permission is given to photocopy this particular page of The National Logo Exchange newsletter for classroom use. The National Logo Exchange, PO Box 5341, Charlottesville, VA 22905 USA.
NLXual Challenges
by
Robs Muir

An XLent Problem

OK. Let's face it. If we were still using Roman numerals, these funny machines we call computers would be very much different, right? Can you imagine the internal architecture? A designer's nightmare! And you thought decimal to hexadecimal conversion was difficult!

AD ROMANUM

Return with us to the thrilling days of Nero et Brutus and relive the wonders of the Roman numbering schemes. Perhaps you will discover why the ancient Romans had nothing in their future; indeed, the zero had to be invented by the Persians.

As you may remember from your 5th grade math book, the Roman numerals are composed of the "numbers" I, V, X, L, C, D, M, and so on, with values represented in our (Arabic) system as 1, 5, 10, 50, 100, 500, 1000, and so on. Using a unique notation, some numbers appear to subtract from one another. For example, IV is 4. Others appear to be cumulative; VII is 7.

NUMERI LOGORUM

This month's problem (come back here!) requires you to use your ancient QWERTY keyboard to compose a Logo program to translate ancient Roman numbers into the slightly less ancient Arabic numbers. The ultimate goal is to permit your Logo computer to add two Roman numbers together and provide the answer in our more familiar numbering system. For example,

ADD "MCMLXXXIV "XVI
2000

Of course, with some initiative, this problem could be extended to include the full range of math functions. (How would you work with fractions?) You also could have Logo return the answers in Roman numerals, and work in a Latin lesson to boot.

ADDERE "XVII "IX
XXVI

While you're at it, try some things like ABANTE "C (a.k.a. FORWARD 100, natch!).

To receive a potpourri of solutions from other readers, send your procedure listing (or your bug-infested attempts) along with a self addressed stamped envelope to:

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

Robs Muir is a teacher in the Claremont CA Unified School District and Claremont Graduate School. He presents Logo workshops in the southern California area.

Logo Some Z's is the newsletter of the Logo Users Club of Japan. Edited by Setsuko Abe and packed with information on the Japanese Logo scene, the quarterly publication costs $30 US per year, and is published in Japanese. Logo Some Z's, Geodesic, Inc., Parkheight #402, 1091 Kumagawa, Fussa-shi, Tokyo Japan.
Logo Computer Systems Inc. (LCSI) announces the following list prices now in effect for their products. Apple Logo and 
Apple Logo II are $99 each. Apple Logo is 
available in Classroom Kits (10 language 
diskettes, a set of manuals, 10 reference 
cards, and a sample and tool kit diskette 
with documentation) for $399. Atari Logo is 
priced at $99 for the complete package, and 
$79 for the cartridge and reference card 
alone. SmartLogo for the Coleco Adam costs 
$74.95. For more information, contact 
LCSI, 220 Fifth Avenue, 
Suite 1604, New 
York, NY 10001, 
or 9960 Cote de Liesse Road, 
Lachine, Quebec, Canada H8T 1A1.

Terrapin Logo is now available in 
packs of multiple copies. A set of 10 
language diskettes and reference cards, a 
utility diskette, and a set of documenta­
tion costs $299.95. A set with 20 language 
diskettes and reference cards costs 
$525.00. Terrapin also has announced a 
second utilities disk for Terrapin Logo for 
the Apple. The package contains 29 pro­
grams, including a driver for the Apple 
color plotter, a picture printing program 
for the Imagewriter, keyboard characters 
for the turtle screen, and a set of low 
resolution graphics tools. The cost is 
$19.95, and may be obtained directly from 
Terrapin, Inc., 222 Third Street, Cam­
bridge, MA 02142. (617) 492-8816.

Jim Muller and the Young Peoples' Logo 
Association (YPLA) announce their 80 page 
Logo resource guide, The Logo Library, a 
listing of Logo versions, articles, books, 
groups, products, and projects. This publica­
tion contains comprehensive information 
on many aspects of Logo. To obtain a copy, 
send $7.95 plus $1.50 for shipping and 
handling costs to YPLA, PO Box 855067, 
Richardson, TX 75085.

It's never too early to think about 
what you will be doing next summer! Why not 
teach Logo at a summer camp? David Sachs of 
Computer Summers, Inc., announces vacancies 
for Logo teachers at some 15 locations 
across the country for the summer of 1985. 
Send your resume and references to David at 
PO Box 436, South Salem, NY 10590.

The New York State Association for 
Educational Data Systems holds its 19th 
annual conference November 18-20 at Kiame­
sha Lake, NY. Beginning and advanced Logo 
workshops are scheduled. For more informa­
tion, contact Eileen Gress, Fairport Cen­
tral Schools, District IMC, 140 Hurlburt 
Road, Fairport, NY 14450.

Vanderbilt University and IBM are 
sponsoring an innovative conference en­
titled "School of the Future" in Nashville, 
TN, on November 8-10. Five discussion 
groups meet simultaneously throughout the 
conference, each led by a panel of experts. 
Linton Deck, conference chairman, suggests 
that Logo teachers and supervisors might be 
interested in the groups on teaching and 
management. Panelists include Karen Bil­
lings, Karen Sheingold, and Roy Pea. For 
more information, write Future School Con­
ference, Peabody College, Box 164, Vander­
bilt University, Nashville, TN 37203. (615) 
322-7311.

The fifth annual Florida Instructional 
Computing Conference will be held January 
29-31, 1985 in Orlando. A preconference 
Logo workshop is scheduled, along with a 
number of Logo presentations during the 
conference itself. For more information, 
write Mr. David Brittain, Office of Educa­
tional Technology, Florida Department of 
Education, Knott Building, Tallahassee, FL. 
32301. (904) 488-0980.

The second annual international con­
ference, "Computers in Education" is sche­
duled for December 12-14, 1984 in Montreal, 
Canada. Featured speakers include Alan Kay, 
and Dan and Molly Watt. For additional 
information, write Computers in Education Con­
ference, GEMS Conference and Consulting 
Services, PO Box 367, Snowdon, Montreal 
Canada H3X 3T6. (514) 735-1388.

The September 1984 issue of Classroom 
Computer Learning contains a stimulating 
Logo article by Pamela Sharp. "Using Logo 
to Shape Artistic Ideas" shows how to use 
Logo in art class. Four separate lessons 
are given for dual art and computer art 
work. CCL, Learning, Inc., 19 Davis Drive, 
Belmont, CA 94002.
George Cathcart, an education professor at the University of Alberta, sends the following procedures for printing inverse letters on the Apple Logo text screen.

TO INVERSE :WHAT
IF EMPTYP :WHAT [STOP]
MAKE "WRD FIRST :WHAT
WRITE :WRD
TYPE CHAR 32
INVERSE BUTFIRST :WHAT
END

TO WRITE :WRD
IF EMPTYP :WRD [STOP]
TYPE CHAR 128 + REMAINDER ASCII FIRST
:WRD 64
WRITE BUTFIRST :WRD
END

Here are some examples to try.

INVERSE [ALL THIS IS HIGHLIGHTED]
TYPE [TRY \ ] INVERSE [THIS] TYPE [OUT!]

Eadie Adamson of the Nightingale Bamford School in New York City sends this procedure to make erasing easier in Commodore Logo. She has used it successfully with her kindergartners.

TO E :DISTANCE
MAKE "COLOR ITEM 4 DRAWSTATE PENERASE
BK :DISTANCE
PC :COLOR
END

Dear Editor,

I think that as teachers become more fluent in Logo and explore its expressive capacities, there may be a broader perception of its classroom applications. Teachers may become more comfortable using Logo as an exploratory medium and less inclined to present it... as a structured, sequentially organized subject.

In this respect, I am personally in disagreement with the positive review of the Logo Guided Discovery Kits in the September 84 NLX. I think one of the glaring deficiencies of this approach is the lack of any single keystroke procedures to permit young children to explore the medium. In fact, I think this kit fosters precisely the wrong sort of implementation of Logo for teachers new to the language.

Robert J. Player
New York City public schools

> Thanks for your comments. The Logo Guided Discovery Kits were intended to be supplemental materials for teaching with Logo. Teachers new to the language need a variety of resources and specific suggestions to set up their learning environments. When combined with concrete activities and explorations, the Logo Guided Discovery Kits can aid creative learning.

School District No. 35 (Langley) has published a 79 page curriculum report which outlines planning for computer education within the district. Seeing the computer as a tool to free students to develop further their learning and thinking skills, the district computer committee based their plans accordingly. The section on the selection of computer languages is especially interesting, and makes a good case for Logo. Copies of this report are available for $10 US. Write School District 35 (Langley), Attn: Patrick McManus, 22259 48th Avenue, Langley, British Columbia V3A 3Z7, Canada.

The Educator 64 from Commodore is billed as "a computer designed for schools." It appears to be a Commodore 64 computer in the PET computer housing, with some additional features. It runs Commodore Logo and all other Commodore 64 software. Check with your local dealer for prices and availability.