THE NATIONAL LOGO EXCHANGE
The Logo Newsletter for Teachers

Volume 4 Number 9 FORWARD 100! May 1986 Page 1

Logo Quilt: A Collaborative Learning Project

by
Dan Watt and Molly Watt

The full power of Logo includes:

- the ability to use top-down planning in designing a project,
- the flexible use of procedures, subprocedures, and variables,
- clear naming schemes that make complex procedures easier to read,
- the use of tool procedures,
- the use of modular, state-transparent procedures, and
- an understanding of how to manage the Logo file system and workspace.

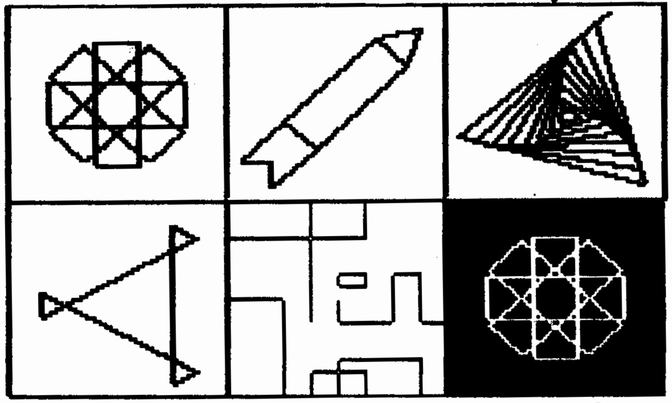
Many Logo students do not reach this level of command over the language, however. We have observed that, as long as Logo learners (adults as well as children) work on individual projects, they tend to solve problems in personal, idiosyncratic ways, and avoid learning these powerful techniques. In our experience, people do not learn to use these ideas until they see a clear need for them.

One way to create such a need is to involve several students in creating a collaborative project, in which each one does a part of the programming, working towards a combined result. Ideally, students should have mastered the basics of writing and editing procedures before starting on a collaborative project.

A collaborative project makes an excellent culmination to a semester's activity. If you are looking for a year-end Logo project, here is an idea to consider.

To Quilt Is To Collaborate

One of our favorite collaborative projects is the design of a Logo quilt. We ask each student to contribute an existing turtle graphics design. The group must then create a superprocedure for the entire quilt, and put all the designs together into an overall pattern. The quilt shown here was made by a group of five students. (Ed. note: Full details of this particular project are contained in Chapter 7 of *Teaching with Logo*, by Molly and Dan, Addison-Wesley, 1986.)



Students who put a quilt together must solve a number of problems that lead them to use powerful programming ideas.

First, they must engage in group planning, in order to put the different designs together.

Second, they have to agree on a strategy for naming procedures and subprocedures so that each subprocedure involved in the project has its own unique name and so that all the procedures can be read and understood by all participants.

Quilt continued

One way to accomplish this is to have each programmer add his or her initials to each procedure name. Dan Watt's triangle would be called TRIANGLE.DW; Molly's would be TRIANGLE.MW, for example. This makes it easy to merge workspaces without duplication, but it can fill up the working memory rather quickly.

If similar procedures appear in several individual designs, the group might agree to use one tool procedure called TRIANGLE to eliminate unnecessary duplication. This may be especially important if the group is using a 64K version of Logo!

The Need for Transparency

The quilt project also encourages students to see the value of state-transparent turtle graphics procedures (procedures that begin and end with the turtle in the same position and heading). This is because the simplest way to design a quilt procedure is to have each subdesign begin and end in the same place - for example, in the lower left hand comer of each square.

The use of variables often comes into play as well, as students adjust the sizes of their designs to fit into the squares of the quilt.

Another feature of a quilt design project is that it can be accomplished by students at different levels of expertise. Each student can contribute a design, whether simple or complex. Less advanced students leam more sophisticated programming techniques during the project. More advanced students learn something about the thinking of their less sophisticated partners while appreciating their contributions to the whole project. (This makes quilt projects especially useful for teacher workshops.)

In addition, we observe that "real world" programming projects are more often accomplished by groups of collaborators than by individual programmers. If one of our goals for using Logo is to promote computer "literacy," this gives us another reason to encourage collaborative programming.

Turtle Tradition

We also like the fact that quiltmaking is a traditional art form, with techniques and aesthetics that make use of different materials and technologies. This shows that computer activities do not have to take place in a high-tech vacuum, but can incorporate ideas from other ages and cultures.

Some people go the other way, and transfer Logo quilt designs to cloth for embroidery. At our 1985 Logo Institute, students chose a Logo quilt design as a symbol for the Institute T-shirt, bringing the image full circle. An idea began on fabric, moved to a computer, came off the computer, and back to fabric in a form that symbolized both the beauty of Logo graphics, and the power of group collaboration.

(l. to r., Dan Watt, Molly Watt, Tony Stavely)

Additional Ideas

Other turtle graphics projects are also suitable for collaborative programming projects. Among the ones we've heard about from teachers is having students design different parts of a face (eyes, nose, mouth, etc.) and put them together to make a unified whole. The same can be done with a stick figure of a person, or an outdoor scene involving buildings, trees, flowers, people, animals, and vehicles.

One group of teachers we worked with made an outline of a sweater on the screen, and wrote an interactive "sweater factory" procedure which used the earlier turtle graphics projects done by all the members of the group as designs to decorate the sweater. The same important programming ideas can emerge from any of these projects.

Getting It All Together

Although Logo is known as an "easy" computer language for both students and teachers, we believe that teachers play critical roles in creating learning environments with Logo. Students cannot "discover" everything for themselves. Their teachers must be prepared to coach them through problems, to provide structures, tools, and models, and to know what areas provide the most powerful potential benefits.

Logo has many powerful problem solving ideas built into it. Students might not discover these ideas while working on individual programming problems. Suggesting a collaborative programming challenge is an effective way to help students encounter these important ideas. Why not try it and help your students get the most out of their Logo programming experiences?

Dan Watt and Molly Watt are teachers, lecturers, authors, and consultants in the field of educational computing. Write to them in care of Educational Altematives, Gregg Lake Road, Antrim, NH 03440.

From The Editor

by Tom Lough

Turtle Awareness

If all goes well, a turtle will raise her head above the waters near Padre Island in the Gulf of Mexico sometime this month. She will sense a special feeling about the stretch of beach she sees, and swim to shore. Soon, joined by others, she will search for a special location in the sand to dig a bell-shaped chamber, and deposit about 100 leathery eggs.

The special significance about this event is that it will be the first time that a Kemp's ridley turtle *(Lepidochelys kempi)* has ever nested anywhere except a particular small section of Mexican beach. Until recently, the nesting turtles have been at the mercy of poachers and coyotes; the population was driven nearly to extinction.

A few years ago, however, supported in part by a grant from the Exxon Company USA, a project to save the ridley turtle was begun in cooperation with the Mexican government, the National Park Service, the National Marine Fisheries Service, several other federal organizations, and an enthusiastic volunteer organization called Help Endangered Animals Ridley Turtles (HEART, PO Box 681231, Houston, TX 77268-1231).

Each year at nesting time on the Mexican beach, under the protection of armed Mexican troops, ridley turtle eggs were collected as they were laid and swiftly transported to Padre Island. There, the eggs were cared for by interns until they hatched. The baby ridleys were released on the Padre Island sand. Interns stood by as the turtles scurried toward the surf.

After a few moments in the water, the turtles were carefully recaptured and placed in special holding tanks for a year, fed by HEART volunteers. Then, being large enough to have a better than average chance at survival, they were tagged and released for good.

It is hoped that being hatched at Padre Island and crawling across the sand to the Gulf of Mexico waters has imprinted that beach as their future nesting beach.

It is in 1986 that the first significant number of the transplanted ridleys will be sufficiently mature to nest. If the experiment is a success, then the ridley turtle will have a greater chance of survival, with two nesting beaches in operation. We'll know the answer soon.

If you are like me, your level of "turtle awareness" has been heightened by your Logo involvement. You read with interest stories such as that about the ridley. If you see a turtle in the middle of the highway, you stop and rescue it. You collect turtle objects, and now sport a collection of at least a dozen plastic, ceramic, and glass turtles, plus a number made from materials such as walnut shells, and nuts and bolts. (Posy even gave me a topiary turtle!)

This awareness affirms to me once again the richness of the myriad interconnections of the learning process. Interest in one subject can lead to so many others. Time and again, when I observe people using Logo, the learning and experiences lead down unanticipated trails. Exclamations such as, "Hey, this is just like what we learned in ..." are commonplace.

Let's take a lesson from our turtle awareness, and try to be more sensitive to the thousands of additional learning directions Logo can make possible. To be sure, many of them are not in the curriculum. But, like the ridley, each of them is important.

FD 100!

Tom

PS: The mailbag is one of the most important fixtures here at the NLX. This summer, I hope to have time to read. Would you take time to write? Please let me know "how you like your eggs (X)." Is the NLX meeting your needs? What can we do to improve your newsletter?

Tipps for Teachers

by Steve Tipps

Goaltending, Year's Ending

The end of the year can be a time for reflection both for teachers and for students. In the rush to finish chapters and give tests, taking time to ask the really important questions is difficult.

- What have we accomplished?
- What have we learned?
- What can we do now that we could not do in September?

These questions seem essential in the Logo class because it is seen as a place where accomplishment is recorded in terms of personal growth.

Schools, however, generally report accomplishment in percentages right and wrong, grade level equivalences, and standard scores. This conflict between Logo evaluation and school evaluation echoes of the difference between mathematics and school mathematics which Papert discusses in *Mindstorms* (BASIC Books, 1980). This difference is one which Logo enthusiasts must address.

Evaluation Colorado Style

The issue of evaluation served as the basis of a long conversation recently with Forrest Smith of Jefferson County (CO) Schools. Jan Bybee, Kathy Nutting, and Forrest co-ordinate one of the largest Logo projects in the country.

Tipps continued

They want to describe the benefits of Logo they see in the classroom without interferring with the excitement and energy which students and teachers show. They also want to know which classroom practices foster learning with Logo. Putting down on paper what they see every day is the problem.

Goaltending

The first step in evaluation is to be clear on intended goals. Knowing goals is essential in determining what to evaluate. On this point, Logo is a more than a little slippery. It is so flexible and and malleable that educators see different outcomes with Logo. The goals stated for Logo are many:

a. Computer awareness
b. Programming
c. Exploration of ideas in mathematics, English, and other subject areas
d. Problem solving
e. All of the above

The general question of goals with computers in school has often been overlooked in the rush to get equipment yesterday. However, Logo has been subject to scrutiny partially because so many claims have been made for it. From the classroom perspective, the value of Logo can be assessed according to goals of both the teacher and the student.

Computer Awareness Goals

In computer awareness, a teacher might look directly at use of the computer with a checklist.

1. Load a disk
2. Use editing functions
3. Save a program
4. Use keyboard function keys
5. Care of disks and computer
6. Etc.

Obviously, these same items might also be evaluated using word processing, the Koala pad, or other kinds of software. In keeping with the cooperative philosophy of Logo, both the teacher and students keep a checklist and compare their assessments.

Programming Goals

Some people believe that teaching programming is the primary goal of Logo. Many, if not most, teachers begin working with Logo as an "easy language" to teach programming. If programming is the goal, then the evaluation method will involve use of Logo to write programs, to do tasks, and demonstrate skill with the language.

The sophistication of problems and programs can range from "Draw a five-sided figure," to "Write a program which alphabetizes names and averages a list of grades for each person," and beyond.

In the first case, the simple turtle commands are needed with a REPEAT structure. In the latter, lists, functions, recursion, and subprocedures are needed.

Even more specific problems can be written for demonstration of specific programming commands or concepts. For example, "Use input variables in a procedure to calculate interest on a savings account," or, "Write an interactive program which surveys your friends on their favorite foods."

The means of determining whether the programming goal is accomplished is also a checklist.

1. Did the program accomplish the task?
2. Did the program use commands correctly?
   FORWARD, BACK, RIGHT, LEFT
   REPEAT
   etc.

Although many teachers begin with the goal of programming, most soon find that Logo is not so "easy" and that their real goals lay beyond programming for programming sake. The commands were only instrumental to other goals.

One of my goals for students is to alert them to "good habits" in computer use. Some of the good habits are oriented toward programming, but most have wider application and probably belong in the awareness category. Each project includes these particular points for evaluation.

1. Readability - Is the printout organized so that the program can be read and understood? Are the procedure names and variable names meaningful and understandable?

2. Modularity - Does each procedure do a task? Are the procedures and subprocedures organized?

3. Documentation - Are remarks used when needed? Is there a title page in program? Is there a procedure tree drawn? Are pictures provided as needed?

4. Saving - Is the file saved on the class disk? Is the file clean of any extraneous procedures?

5. Operation - Does the program work, or have bugs been isolated?

The checklist is given to students as a way of reinforcing the "big" ideas to consider while working on their Logo projects. Whether students go on with programming or with applications, they are alert to some good habits.

Good habits and language can be evaluated in the short range on a project; however other benefits of Logo programming may also need to be considered. One of the major payoffs for Logo is likely to be in the Logo-to-Pascal link. Although I had Fortran and BASIC, I was a Pascal dropout several years ago.

Tipps continued

This semester, I am working with Pascal again and am finding it so much easier ... almost intuitive. I also see the frustration that students have with concepts of modularity, parameter passing, naming, and structures. They seem to concentrate on the placement of semicolons rather than on structure. These students have more programming experience than I, but their experience does not seem to help them think in Pascal. I believe that Logo has made the difference for me, and that every beginning Pascal course should start with Logo.

Exploration Goals

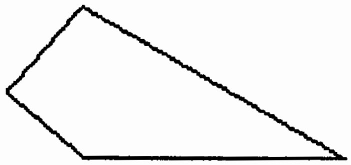
A different goal with Logo is as an exploratory tool in subjects. The natural turtle world holds many fascinating ideas about arithmetic and geometry. Some Logo pioneers have developed tools to explore other content ideas.

Glen Bull and Paula Cochran emphasize language in their "Listful Thinking" column in the NLX; *Microquests* by Kathleen Martin and Donna Bearden (MBI, 2502 Peach Tree Lane, McKinney, TX 75069) is a must for anyone involved in intermediate mathematics. Tom Lough and Jim Hurley have worked with Logo and physics explorations. (See "Logo and Physics" in the January issue of *The Physics Teacher*, and *Logo Physics*, published by Holt, Rinehart and Winston, 1985.)

Whatever the content you are interested in, the evaluation method must match content goals. The conflict between the philosophy of Logo and demands of school subjects might be at its fiercest in this situation; however, this need not be so. No teacher should be comfortable with using Logo, and Logo alone, to explore the rule of 360. Geoboards, string art, protractors, compasses, and the textbooks are also instructional aids for finding out about interior and exterior angles.

Logo is one of many instructional approaches which might be used to explore and learn a concept. But the teacher is the most important element in creating an atmosphere of exploration.

After Logo and many other methods are used and the teacher has had children collect their discoveries, questions on the math test about 360 degrees, size of angles, and the sum of the angles are perfectly appropriate. A question might even pose a Logo-like situation in which an ant was travelling around an irregular quadrilateral and request an estimate of the number of degrees the ant would have to turn at each corner.



The objectives involve degrees and angles and shapes; the evaluation should also. With each content goal, questions can be posed. However, the "fact" questions do not fully assess the benefits of using Logo tools. Part of the result is affective, and that affective dimension should also be addressed with attitude questions. An example appears in the "Logo Teacher Feature" in this NLX issue.

Part of the result is also learning empowerment. A researcher might use the Nowicki-Strickland Locus of Control to assess this (See, for example, Nowicki, S., and Strickland, B. "A Locus of Control Scale for Children." *Journal of Consulting and Clinical Psychology*, 40 (1), 1973, pp. 148 - 154.), but teachers can observe and interview to assess self direction and goal orientation. Such measures may be too "soft" for statistical reports, but they are vital for giving the complete picture of Logo exploration.

Problem Solving Goals

Problem solving behavior is one of the most fascinating constructs in psychological research, and also one of the most difficult to describe, document, and interpret. Is problem solving ability specific to a problem situation or type of problems, or is it generalized as an approach? What are the characteristics of good problem solving, and can they be taught or caught? Although problem solving was one of Logo's greatest selling points, documenting and evaluating this outcome has been difficult.

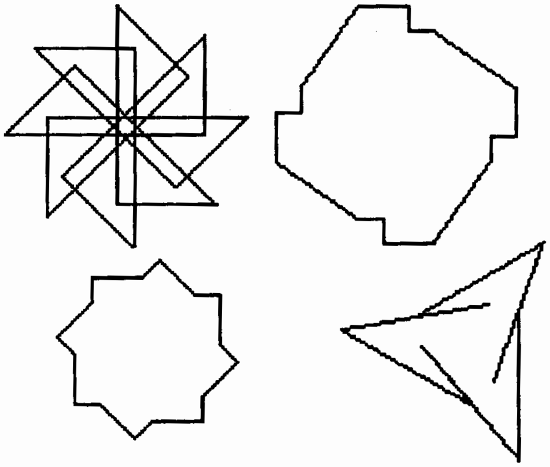
Despite the difficulty, I have ways with which I am content, if not satisfied, for evaluating problem solving. One way deals with planning a task. Logo users seem to look at problems in a more "bite size" fashion.

For evaluation of this modular thinking strategy, I have asked students to describe the major parts of a task. Sometimes this is a turtle picture which can be decomposed. Sometimes, the task is an everyday problem, such as how to rearrange the room or complete some class project. Instead of working out all the details first, students look at major tasks. Logo provides a common experience to talk about the strategy adopted.

Another problem solving technique is anticipation of results. Can students decompose a problem and see what is going on? A method which is specific to Logo learning is analyzing statements and guessing what they do. REPEAT statements like those in the REPEAT GUESS exercises in *Nudges: Apple Logo Projects* (Holt, Rinehart and Winston, 1985) are good examples of this technique.

Can you match these statements with the pictures on the next page?

```
        REPEAT 8 [ FD 20 RT 45 BK 40 RT 90]
        REPEAT 8 [ FD 10 RT 45 FD 20 LT 90 ]
        REPEAT 4 [FD 20 RT 90 FD 10 LT 90 FD 10 RT 35 FD
40 RT 55]
        REPEAT 3 [BK 40 RT 140 BK 40 FD 20 LT 20]
```



Evaluation ... and Grading

Evaluation *is* important and difficult, but it is essential in the implementation and improvement of Logo use. Begin with the goals which you have with Logo, and keep evaluation hooked to the goals. Unless you are teaching computer science with Logo, look on Logo as instrumental rather than as the final product. Use evaluation techniques such as anecdotal records, project files, and demonstrations. Checklists and exercises which demonstrate growth in specific areas are a valuable adjunct to demonstrations and files. Include student evaluation as an important element. Avoid grading Logo per se wherever possible. (Thank heavens, Logo does not appear on report cards yet!)

"If I were king for just one day ... " I would do away with grades completely in schools and concentrate on evaluation. Evaluation and grading are related but not the same. Evaluation asks the important questions about learning; grading assigns values to learning, often in an arbitrary manner.

I would like to be clean on the issue of grading Logo. As many of you, however, I have the onerous job of grading courses. I attempt to reconcile my personal dilemma with very specific expectations. I list exactly which commands and which good habits are to be in a project, with a specific number of points for each. However, the projects themselves are completely open-ended, and take on many forms.

The gap between present behavior and belief distresses me. I invite your comments about these ideas. Write to Steve Tipps, Midwestern State University, 3400 Taft Boulevard, Wichita Falls, TX 76308, or send email on CompuServe to 76606,1623.

Have a wonderful summer!

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 Teacher Feature

by Rebecca Poplin

Featuring: Sue Wolff

A few years ago, the sound of pounding hammers and the smell of sawdust signaled that kids were learning in Sue Wolff's Fairfax County (VA) classroom. Her fifth grade students constructed their own special buildings to use in the classroom for different purposes, and learned lots about measurement, planning, cooperation, and hard work in the process. Each building measured about 6 feet by 8 feet, and was made of 2x4's and paneling.

From Dust to Disk

Of course, as soon as computers became available, her students constructed a computer building. Sue realized that the classroom had filled up with buildings and that the carpentry dust wasn't too good for the computers. It was, "Out with the dust, and in with the disks," as the new technology began to replace the old.

Five years ago, Sue began borrowing computers to use in her class. Having started as many of us did with BASIC, she moved first to PILOT, and then to Logo. Fascinated with the types of thinking she observed, Sue wanted to provide her students with a more intense involvement with Logo. Her first step was to write a proposal called Learning Through Logo (LTL). She requested administrative backing for a curriculum which would have computers as the centerpiece of a creative interdisciplinary learning program.

LTL School Room

Before the 1983-84 school year began, Sue organized a meeting for parents of rising fifth graders. Sue explained her project and invited students to participate. That first year, another fifth grade was used as a control group. Sue spent two weeks at the beginning of the year teaching Logo. In the mornings, students participated in content area discussions, research in the library, and teacher-directed instruction, but very little traditional seat work.

Whenever students showed they had studied a topic thoroughly and had lots of information, they would write computer programs in the afternoon. Once or twice a week, students had additional lessons in Logo.

In order to study a topic, students had to submit a proposal. For example, a group or individual might choose a topic like the respiratory system, pick out the most important information, put the information in the computer, and illustrate the report using turtle graphics. Occasionally, the topic had to be narrowed down, as

Feature continued

when a group of students proposed studying "history." Students also wrote math and reading lessons, and made computerized mad libs related to social studies and parts of speech. In the course of the year, Sue's class produced quite a bit of software.

Many of their programming efforts and experiences were documented on videotape which was used for presentations to parents. Sue also showed the videotape as part of her presentation at Logo 85.

Doing Logo Makes a Difference

During the first year of the LTL project, Sue noticed many effects of the intense technology-based study. Some results were nebulous, others more tangible. An immediate outcome was that students produced long stories for their parents when asked, "What did you do in school today, dear?"

An attitude survey showed differences between the LTL students and the control class. Among other effects that didn't show up on tests and surveys, the school librarian began to notice differences in self-direction in Sue's students. Even though the curriculum had not followed the strictest topic guidelines, students in the program experienced more growth in standardized test scores than the other class. During the second year, the program expanded to include the other fifth grade.

By this time, all of Forestville Elementary had gotten involved with computing. As student government fund raisers, students sold purple T-shirts proclaiming, "I'm a computer kid." (Sue is a T-shirt enthusiast. Whenever visitors were scheduled to observe the LTL program, she and all the students wore their special shirts with turtles on them!)

Additional Challenges

Sue has worked as a teacher trainer in Fairfax County and has taught several Logo courses. All teachers have at least an hour a week using the school lab. Some do word processing, others use software, but most choose Logo at least part of the time.

Recently, Sue has taken on a new challenge. She has become a computer resource teacher, working with approximately 45 area schools at all levels. Many of the teachers she helped get started in Logo will get the opportunity to work with her more often.

Sue wants to challenge teachers to use more text with graphics in Logo. She feels that text can help to sell a Logo program, because students can improve their writing and because text helps Logo fit in with the curriculum. With text, students and teachers have a hard copy of something that looks very much like "regular school work." Sue Wolff believes that, when the computers are in place and programs are going, *then* curriculum revision can take place.

If you know a teacher who should be featured in this column, please contact:

Rebecca Poplin
2421 Fain Street
Wichita Falls, Texas 76308

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

Logo Notes

Academia de la Investigacion Cientifica announces the III International Symposium on Computers and Children's Education, to be held October 25 - 31, 1986, in Puebla, Mexico. At last year's symposium, a number of the presentations were on Logo. Here is an opportunity to learn about Logo use from a different cultural perspective without the expense of overseas travel. For symposium information, write to Academia de la Investigacion Cientifica, Insurgentes Sur #949-9, Mexico D. F. 03810. Copies of the proceedings of last year's symposium are available for 3000 pesos from Dr. Jorge Bustamente at the above address. Daniel Orey is maintaining a mailing list of Logo users from the USA and Canada who are interested in attending, and will be working on a group discount for travel and accommodations. Write to:

Daniel Orey
CIIMTE - COE
The University of New Mexico
Albuquerque, NM 87131

MACUL SIGLOGO Formed

During the final session of the recent Michigan Association for Computer Users in Learning (MACUL) conference, a group of thirty-five Logo enthusiasts assembled for the purpose of establishing a Logo Special Interest Group (SIG). Christine Canning chaired the informal meeting with assistance from Reggie Bowden, Karen Bunting, Anne Cairns Federlein, Jim Fry, Lynn Lieberman, and Terry Rudman.

After some introductory discussion, the group voted to take the steps to establish itself as an official MACUL SIG. Tom Lough, editor of *The National Logo Exchange* newsletter, spoke briefly, commending the group for its spontaneous generation and enthusiasm. He encouraged the SIG to disseminate its news and information, as well as to provide a model for other groups across the country.

The SIG then divided into three discussion clusters, to explore SIG objectives, activities, and conference planning.

For more information and membership applications, write to Chris Canning, MACUL SIGLOGO, PO Box 628, Westland, MI 48185.

Listful Thinking

by Glen Bull and Paula Cochran

Turtles, Lists, and Spaceships

In our previous column, we considered a question that Brian Harvey raised for participants attending the Logo 86 conference to be held this summer: "Is it important for the same language to contain both turtles and lists?" We said that we thought it was, and added that very often the turtle can be a bridge which makes the effects of list operations more tangible.

In this column, we would like to look at an example of how the two capabilities can complement one another. Generally, we introduce lists through examples which use words and sentences, since everyone constantly uses sentences in everyday life. Once words are used to introduce the concept of lists, through poetry and conversational dialogs, examples involving lists of numbers can be incorporated.

Computers have the potential for making subjects interesting that would otherwise be boring. You would expect that a math major would be interested in trigonometry. The trick is to make the same subject interesting to someone whose first love is art or history. Papert claims we all have the inherent ability to become interested in subjects such as trigonometry, if they are introduced in an appealing way.

In a recent seminar, we asked all who were interested in trigonometry to raise their hands. Only two people did so (both were math teachers). Everyone else in the seminar had a reaction that ranged from disinterest to downright aversion. We have found this is not unusual.

Math and Original SIN

Yet, trigonometric functions, such as the sine function, can be fascinating if they are explored with the aid of a computer, rather than through the static pages of a book. Even if trigonometry is not your first love, please don't stop reading. Most of the procedures are only three or four lines long, and they are almost identical to the ones we used for Logo poetry in earlier columns. (What we're trying to say is that we have a bridge we'd like to sell you ...)

The SIN operation is the Logo equivalent to the sine function. Every angle has a SIN associated with it. (We know your mother told you to avoid sin. However, this SIN is pronounced "sign," not "sin.") There is a pattern to these sines. This procedure will print the sines associated with the degrees in a circle. (In all of the examples which follow, omit the brackets around STOP if you are using Terrapin or Commodore Logo.)

```
TO SHOW.SIN :DEGREE
IF :DEGREE > 360 [ STOP ]
PRINT SIN :DEGREE
SHOW.SIN :DEGREE + 20
END
```

You can run the procedure by typing this:

```
?SHOW.SIN 0
0.0
0.342020
0.642788
...
```

The SHOW.SIN procedure prints a series of numbers that look rather forbidding, primarily because they are long decimals. It's easier to see the pattern if we change the procedure so the numbers are expressed in turtle steps. We can do this by multiplying each sine by 100.

```
TO MAGNIFY.SIN :DEGREE
IF :DEGREE > 360 [ STOP ]
PRINT ROUND 100 * SIN :DEGREE
MAGNIFY.SIN :DEGREE + 20
END
```

Now the numbers produced are more like ones you might use with the turtle.

```
?MAGNIFY.SIN 0
0
34
64
87
98
...
```

Turtle on a Pogo Stick

In fact, imagine that the turtle was moving up and down from one point to the next. It would start out at 0, move up 34 turtle steps, then go to 64, and so on. Here is a procedure to move the turtle in just that way. We've multiplied each sine by 50 instead of 100 to keep the turtle from going off the bottom of the screen in the splitscreen mode.

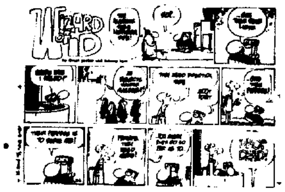
```
TO POGO :DEGREE
IF :DEGREE > 359 [ STOP ]
SETY 50 * SIN :DEGREE
POGO :DEGREE + 10
END
```

We called the procedure POGO because it makes the turtle bounce up and down almost as though it were on a pogo stick. In fact, the turtle behaves very much as though it were on a spring. It slows down at the end of the motion and speeds up as it moves through the middle. Is this just an optical illusion? Try it, and see what you think.

```
?REPEAT 3 [ POGO 0 ]
```

The Turtle Seems Listless

Now that you've got an idea of what a SIN function can make the turtle do, let's add new capabilities by drawing upon the power of lists. Thus far we've been asking the computer to calculate each sine over and over again. The time required to make each calculation slows the turtle down a little. In fact, its movement is halting and, well ... just a little listless.



By permission of Johnny Hart and News America Syndicate

We can make the pogo stick bounce faster if we calculate the sines just one time, and store them in a list. Of course, you could write down the list of all the sines used and then type them back in with a MAKE statement. However, it's easier to let Logo do the typing for you. Here is a procedure to store the sines used by the turtle in a list:

```
TO CREATE.SINES
MAKE "SINE.LIST [ ]
DO.SINE.LIST 0
END

TO DO.SINE.LIST :DEGREE
IF :DEGREE > 359 [STOP]
MAKE "SINE.LIST SE :SINE.LIST SIN :DEGREE
DO.SINE.LIST :DEGREE + 10
END
```

Now type

```
?CREATE.SINES
```

Logo will go away for about 30 seconds while it makes the list. Although this seems like a long delay, it's probably faster than you could type the numbers in.

Inspecting the List

When Logo finishes the CREATE.SINES procedure, you can see if it worked properly by printing the contents of the list.

```
?PRINT :SINE.LIST
0 0.173 0.342 ...
```

The numbers are printed across the page in a rather messy fashion. A number of columns ago, we demonstrated the use of the INSPECT procedure to inspect Christmas lists, grocery lists, etc.

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

We can use the INSPECT procedure to make a more careful inspection of the list of SIN numbers.

```
?INSPECT :SINE.LIST
0
0.173
0.342
...
```

We also wanted to use the INSPECT procedure here because one small modification will allow the turtle to use the list of SIN numbers to create the same pogo stick effect presented earlier. However, now the turtle should move faster and more smoothly since there are no longer any delays while the sines are calculated.

```
TO BOUNCE :LIST
IF :LIST = [ ] [ STOP]
SETY 50 * FIRST :LIST
BOUNCE BUTFIRST :LIST
END
```

At this point we are using the familiar INSPECT procedure to perform an operation on a list of numbers instead of words or sentences. The only difference is that we are replacing the PRINT command with the SETY command to move the turtle. See if the turtle moves more smoothly when a list of numbers is used. Its performance should no longer be as listless when you use the BOUNCE procedure.

```
?REPEAT 3 [ BOUNCE  :SINE.LIST ]
```

Springtime

It is clear that the springiness of the turtle is related to the SIN function. Consider other ways that the turtle could move. It could move forward at a constant speed and then stop instantaneously. Here is a procedure that makes the turtle do that:

Listful Thinking continued

```
TO FLIP
FD 50 BK 50
END
```

Compare these two ways of moving the turtle.

```
?REPEAT 3 [ FLIP ]
?REPEAT 3 [ BOUNCE :SINE.LIST ]
```

(This is a visual concept. You'll have to run the procedures to see what we mean.)

The Plot Thickens

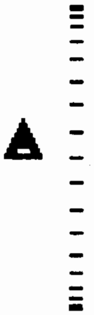
The INSPECT procedure prints the numbers, but the turtle can be used to plot the numbers. The PLOT procedure is just like BOUNCE, except that it leaves a dash at each point along the turtle's path.

```
TO PLOT :LIST
IF :LIST = [ ] [ STOP ]
DASH
SETY 50 * FIRST :LIST
PLOT BUTFIRST :LIST
END

TO DASH
PD SETX XCOR + 2
PU SETX XCOR - 2
END
```

Now type

```
?PLOT :SINE.LIST
```



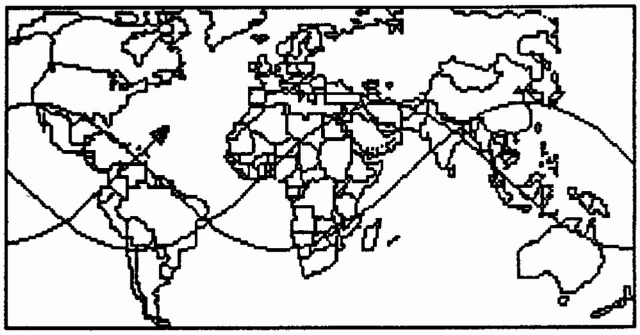
The dashes left by the turtle immediately make it apparent that the turtle really is moving a greater distance between points as it goes through the middle. One of the seminar participants had suggested that the faster movement of the turtle through the middle was an optical illusion. However, the dashes made it clear that the turtle really was moving faster in the middle, and slowing down near the top and bottom.

This illustrates how the dynamic movement of the computer can make concepts clear that are difficult to grasp on a static printed page. (In fact, if you didn't quite follow our explanation, we encourage you to go to the computer and run these procedures.) In addition, the concept is easier to grasp because the computer performs the mechanical drudgery of calculating the points. Even if you were to perform the same task with pencil and paper, the tedium of doing the calculations by hand would take the fun out of it.

Another concept of learning emphasized in *Mindstorms* is the idea of connectedness. It's easier to understand an idea if you can relate it to something you already know. One student confessed that she had plotted (did she mean plodded?) through hundreds of SIN calculations without ever understanding that there was a relationship between this function and springiness. (The actual term is "simple harmonic motion", but rather than let these words get in the way of our understanding, we will simply say that we are talking about the kind of movement represented by a spring or a swingset or a pendulum. For example, think of how a swing slows down at the top of its arc, and speeds up as it goes back through the middle.)

Turtles and Spaceships

This same kind of motion is also associated with many kinds of circular movement ranging from the movement of a bicycle wheel to the orbit of a spacecraft. Possibly you can visualize the path that a spacecraft takes as it moves around the earth ... perhaps you have seen this on a telecast of a NASA mission. Very often, the path of the spacecraft is shown as it moves across the United States, Australia, and other countries.



With a slight adjustment, the turtle can plot a similar path. The only difference between the PLOT and the PATH procedures is that we will move the turtle over about four steps after each point is plotted.

```
TO PATH :LIST
IF :LIST = [ ] [ STOP ]
DASH
SETY 50 * FIRST :LIST
SETX XCOR + 4
PATH BUTFIRST :LIST
END
```

The PATH procedure shows the S-like path that a spacecraft will take as it moves across the world map.

```
?PU SETX -139
?REPEAT 2 [ PATH :SINE.LIST ]
```

To see what happens with successive orbits, clear the screen and try a larger number of repetitions.

```
?PU SETX -139
?REPEAT 5 [ PATH :SINE.LIST ]
```

Listful Thinking continued

If you are interested in pursuing the idea of turtles and orbits, "Kepler" by Jim McCauley is a classic article on this subject. (*The Computing Teacher*, December/January 1983-84, p. 15-22.)

Summary

There are several points that we would like to make. The first is that the transition to use of lists can be fairly natural and painless. The second is that turtles and lists are natural partners. The third is that computers can be used to explore a variety of subjects in ways that are difficult to pursue with pencil and paper alone. The best way to explore a dynamic concept is with a dynamic medium.

We are concerned about the ways that computers and Logo are being used in schools. Teachers very often are being required to teach Logo to fulfill "computer literacy" requirements rather than using the computer and Logo to do a better job of teaching traditional subjects.

Paul Goldenberg has introduced the idea of the "least intrusive" addition of the computer to the classroom. This means that the computer should be used to allow a good teacher to teach more effectively -- but without disrupting her previous teaching style. ("Learning to Think Algebraically: Word Problems and Data Flow." E. Paul Goldenberg, *Proceedings of the World Logo Conference*. Paradise Valley, B.C., October 25-27, 1985.) That is, the computer simply becomes another tool (although a very effective one) rather than a new subject to be learned.

We have expressed this idea as the concept of "using Logo to teach, rather than teaching Logo." The title of this column has been "Listful Thinking," but it could as easily have been "Using Logo to Teach without Years of Study or a Degree in Quantum Mechanics." Lists are just one of many elements in Logo which make it possible to do this.

We wish you a listful summer!

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. His CompuServe ID is 72477,1637. Paula Cochran recently completed the requirements for her doctorate at UVA's Curry School of Education. She is interested in Logo applications in language arts and speech-language pathology.

LogoWriter Workshops and Classes

If you are planning a course or workshop on LogoWriter, and desire assistance with publicity, course outlines, and other support materials, write to Michael Tempel, LCSI, 555 West 57th Street, Suite 1236, New York, NY 10019, or call (212) 765-4780.

TurtleTips

by Jane Toth
and Donna Lanyi

Since our drawing with Logo is just "sailing" right along, it is time to add another challenge. Those who have requested our monthly procedures realize that we usually included a printed message below the picture. Besides adding words to graphics, the use of the PRINT command opens new avenues for our students.

The Turtle Teacher

We always enjoy adding a "cute-sie" phrase to each of our monthly procedures. We do this by using the SETCURSOR command along with a PRINT statement. Here are some ideas for using this technique.

Involve your class in a group comic or picture book. Each student or group could be responsible for completing a picture and a PRINT line to accompany it. Students could write letters, poetry, vocabulary sentences, etc., using the PRINT statement, and illustrate them with a Logo graphic.

We decided to pick topics that we were studying in class from various subject areas - horses from reading, human body systems from health, and US states from social studies. This is definitely a long range project which enables students to work towards a given date.

A useful strategy for working with the Logo text editor and the PRINT statement is detailed in Glen Bull and Paula Cochran's "Listful Thinking" column in the February 1986 NLX.

Also, the exciting new LogoWriter from Logo Computer Systems Inc. (LCSI) would be a natural choice for projects such as this.

(We have heard about people using the Logo editor as a word processor. We have never tried this, but, if a commercial word processing disk is not available in your school, this might be an option to explore.)

From the Mail Box

Phyllis Richardsworks as a computer teacher in the Elmont (NY) Schools. Phyllis sent us a suggestion she found most helpful after reading and using information in Glen and P's excellent "Listful Thinking" column in the December 1985 NLX.

In using their "Instant Tools," Phyllis encountered a problem with the procedure used to quit the program. The children were to press the Q key whenever they wanted to stop. She found that her children lacked keyboarding experience, were easily distracted, and would often press the Q key by accident. Phyllis changed the quit command to a shifted character, requiring the deliberate use of the shift key.

TurtleTips continued

```
TO $
PRINT [ QUIT INSTANT ]
PRINT [ TYPE DRAW FOR REGULAR LOGO ]
TOPLEVEL
END
```

This procedure has eliminated any accidental quitting or other disappointments. Phyllis also included the use of the PRINTPICT command in her Instant, so that she can help her students easily produce printouts of their work. PRINTPICT is in a file on the Utilities II disk for Terrapin Logo, and requires an Imagewriter. She placed the command inside another procedure which uses the shift key.

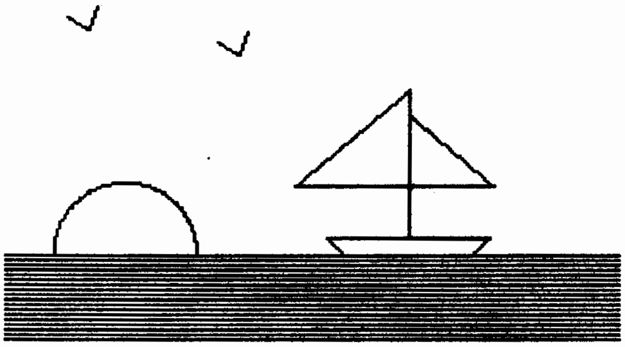
```
TO !
PRINTPICT 1 0
END
```

Thank you, Phyllis, for sharing these ideas!

Procedure of the Month.

During May, as students' minds drift toward summer, use their creative thoughts to design a spring or summer bulletin board. Have students design their own sailboats or hot air balloons. Add some poetry or creative writing stories and you have a great eye-catching bulletin board.

If you are interested in a copy of our procedures, please send us a stamped self-addressed envelope.



We have enjoyed working with you this school year. Special thanks to all those who wrote to us. Have a great summer, everyone! Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691.

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

NLXionary
A Lectionary of Selected Logo Readings

with Commentary and Opinion by Griff Wigley

"An Interview with Seymour Papert," by Fran Reinhold, *Electronic Learning*, April, 1986.

>In case you hadn't heard, Seymour Papert has a new Logo. And when Papert speaks, the educational computing magazines listen. Although both of the following articles on LogoWriter are authored by Papert, the philosophical underpinnings he discusses are substantially different.

"The Next Step: LogoWriter" by Seymour Papert, *Classroom Computer News*, April, 1986.

>Papert discusses a "first tier use" of a technology, i.e., people using technology to do better what has been done all along. The use of computers for the drilling of basal reading skills is an example of a first tier approach; it improves the delivery of an otherwise unchanged curriculum. Word processing, surprisingly, is also a first tier use; it is nothing other than a high powered typewriter (a use he emphatically supports).

>Logo's turtle graphics is a second tier use of computers because it presents a distinctly different approach to math from the traditional curriculum. Papert says that LogoWriter is a second tier successor to writing with a word processor. It allows students to manipulate text in a natural fashion, just as Logo allowed the manipulation of graphics. It is a programmable word processor.

>By allowing students to actually change and / or create new word processing commands, Papert remains true to his Logo philosophy of putting the student in control over the computer. (This feature sounds like a "macro" capability, common to many of the more high-powered application packages.)

>LogoWriter has the ability to combine text and animated graphics, as well as to control the position and "tempo" of the text. This offers the student a medium to work with that's between text and speech, Papert says. It's a tool for any wide-eyed story-telling child who needs help putting the same feeling into his or her writing.

>Papert hopes that teachers who are more drawn to verbal or dramatic content rather than mathematical subhect areas will now have access to the Logo culture. And as for the "transfer" question, he's convinced that "your imagination as a teacher will show you how to use LogoWriter programming as a transition to pure writing."

NLXionary continued

"New Views on Logo," by Seymour Papert, *Electronic Learning*, April, 1986.

>Papert's approach towards explaining LogoWriter in this article is to compare the learning styles of two children.

>Jessie is a planner, attracted to precision and detail; Jessie could become deeply involved with the original Logo in part because its precision-minded turtle lends itself to planning.

>Pat is your stereotypical artist, drawn to color, motion and a plan-as-you-go style of working. Logo in most of its forms is more limiting to a student with this style.

>LogoWriter is Papert's solution to providing a richer environment for the Pats of the world, while still accomodating the Jessies. It continues to provide a microworld of graphics while adding a microworld of text to explore.

>Logo's extensibility allows the creation of new commands by writing procedures for the manipulation of turtle graphics. LogoWriter's extensibility allows the creation of new word processing commands for the manipulation of text. Papert maintains that a word processor should not be an unchangeable "black box" because then *it* is the master, not the user.

>LogoWriter sounds like an improvement over other attempts to combine text and graphics (Kidwriter, Story Machine, Story Maker, Bank Street Story Book, etc.). Its text programmability feature seems the most exciting. Look for reviews of this software in virtually every educational computing publication, including this one.

>One final note: what assumptions might you make about yourself as to your presumptions of the sex of Jessie and Pat? You will find some excellent food for thought on this matter in Sherry Turkle's *The Second Self* (Simon and Schuster, 1984).

If you would like to respond to this 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 facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

Logo Teachers Wanted!

Interested in teaching Logo at a camp this summer? If so, then send your resume with references to Computer Summers, Inc., PO Box 436, South Salem, NY 10590. Camps are located throughout the USA, so specify your geographical preferences..

Logo Disserts:
Dissertations Dealing with Logo

by Barbara Elias

A thesis by Linda E. Morecroft, titled *A Relative Motion Microworld*, presents an example of a Logo microworld designed to teach concepts in relative motion to students in mathematics and physics. Its purposes are to enhance student understandings of relative motion concepts and to provide a rich medium for representing motion in a variety of settings.

Relative motion problems deal with how moving objects appear in varied reference frames and from varied perspectives. For example, the path made by a coin which falls from the hand of a person running across a yard and the way in which pilots determine where to drop relief packages to specified targets on the ground involve relative motion.

Traditional approaches to teaching this topic generally utilize static representations. Such representations are limited for providing students with the kinds of tangible experiences available through computer simulation.

Four basic steps were used to set up the motion simulation:

a. Independent objects were created with the capability of moving on the screen

b. Each object was then told how to move

c. One object was chosen as the reference frame object

d. All objects were set in motion

Problems and explorations in rectilinear and non-parallel rectilinear motion, circular motion, and motion under gravity were prepared and presented in a student text. A teacher's guide was also prepared. The program was developed with Terrapin Logo and used on a 64K Apple II computer. It was also developed in LCSI Logo and used on the Macintosh 512K computer.

The microworld was tested with two tutorial groups of 2 and 3 students each and with a class of 22 students, most of whom were 12th grade physics students. A screening test was given as a pre- and post-assessment. Personal observation, discussions with students, and information from a student survey provided data for the study.

Students spent about one and one half weeks using the materials during their regular school day. The researcher assisted students as necessary. Valuable feedback helped to refine and improve the materials. Most students found the materials helpful, and agreed that their understanding of relative motion had been improved. Post-test scores indicated improvement and students enjoyed using the computer in physics class.

Logo Disserts continued

Physics and mathematics teachers may find the Student Text and the Teacher's Reference Manual especially useful for teaching the topics. In addition, a number of suggestions have been included for teachers interested in developing their own computer based leaming environments.

>>>Morecroft, Linda E. (1985). *A Relative Motion Microworld* (unpublished masters thesis), Massachusetts Institute of Technology, 1985.

(Ed. note: Barbara Elias is accepting copies of recent research results for the NLX Dissertation and Thesis Repository in preparation for the 1986 - 87 publishing year. 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 VA.

Q and A

by Jim McCauley

Q. I heard someone say the other day that Logo doesn't have a "null." What does that mean? Does Logo need one?

A. "Null" is one of those words that has many meanings in the world of computing. For example, there is an ASCII character called "NUL" which is ASCII character 0. Ordinarily, it is a character that cannot be printed, since it has no defined printed character associated with it. It is used as a control code. Your version of Logo may print out some character for it anyway; try typing PRINT CHAR 0 and see what you get.

Plenty of Nothing

The usual meaning of the word "null" is "empty; containing nothing." In most versions of Pascal, *null* means two things: a *null statement* (that is, a statement that means and does nothing) or a *null string* (a string containing no characters).

It may seem silly to have statements and strings that are empty, but they are very useful. For example, if you are writing a data-entry routine, it's handy to define the act of typing nothing and simply pressing the RETURN key as a "null string," which could be used to indicate that you are finished entering data.

Logo has two predefined "empty objects." The empty list ([ ]) is enormously useful as a end-test for recursion. It can be used as a means of detecting if a procedure has "BUTFIRSTed" a list down to the point where it contains nothing. The empty word (") can be used for similar purposes in procedures that are recursively decomposing words.

In many versions of Logo, if you type SHOW READLIST and press RETURN twice, you will see that the empty list is defined as Logo's standard "null object". (MIT Logo users, you will get similar results if you type REQUEST and press RETURN twice; you should get RESULT: [ ] as a response.)

NULL and NIL

The use of the empty list as a null object is part of Logo's heritage from Lisp. There is a common Lisp function called NULL that is ordinarily used to identify empty lists (like Logo's EMPTYP or EMPTY? operation).

The NULL function illustrates an interesting difference between Lisp and most versions of Logo. Logo uses two special words (TRUE and FALSE) to identify Boolean conditions. Lisp has two similar words (T and NIL), but these Lisp words have special extensions to their meanings. In Lisp, any empty list is defined to be NIL (false), while anything that is not empty (a word or a list containing one or more objects) is defined as T (true).

This may sound crazy, but it greatly extends the power of Lisp. It means that any object in Lisp can not only contain data but can also have a true / false value associated with it. This turns out to be a very useful property when doing advanced programming, especially in creating databases or artificial intelligence systems.

When Logo was first designed, there was some discussion about whether or not to retain this Lisp feature. At the time, it was felt that it would be wiser to use only the literal words TRUE and FALSE to indicate Boolean conditions. It was hoped that this would give Logo greater clarity and simplicity without sacrificing anything important. Some recent versions of Logo (ExperLogo and TLC-Logo) have restored the Lispish view of "truth."

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

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

Brian Harvey's Second Volume Announced

The second volumn in Brian Harvey's Logo series, *Projects, Styles, and Techniques*, from the MIT Press, will be available at the Logo 86 Conference.

NLXual Challenges

by Robs Muir

Objective Orientation

The chief problem with writing a monthly article such as this is the everpresent danger that the Fun (with a capital "F) of Logo will vanish in the glare of the DEADLINE - somewhat like the student who hasn't finished enough Logo task cards by the end of the term. I am happy to report that this month's challenge has me more excited about Logo's potential than I was when I first booted Terrapin Logo (version 1.0) on my Apple II!

Among the cognoscenti in computerdom, object-oriented programming is a current fascination. If you haven't read much about object-oriented programming, there is a very simple reason; there aren't many languages that support it! But this style of programming is emergent and may have strong implications for the next generation of languages.

What Is an Object?

Quite simply, an object is computational data structure that behaves like a computer within a computer. One object you are already familiar with is a turtle; you can send it messages, such as FD 50 RT 90, and the turtle-object will perform the task without you explicitly calculating where its new position will be. One "talks" to a turtle without regard to how it computes the result. A turtle uses its own *methods* for interpreting a *message.*

An object could just as well be a user-defined procedure that could return a value or respond to a *message* sent to it. Object-oriented programming allows for such an environment. One need not know how an object "does its thing;" each object represents a much higher degree of modularity than does a simple procedure. This modularity is vital to modern programming and may influence the forthcoming multi-processing computer architectures of the next decade.

The ideas for object-oriented programming were first supported by a language called Simula, developed by Alan Kay. Later, the concept was developed principally during the SmallTalk project at Xerox's Palo Alto Research Center (PARC) (see Goldberg, A., and Robson, D. *SmallTalk-80: The Language and Its Implementation.* Addison-Wesley, 1983). At the heart of SmallTalk is the concept of "message passing," where one object may transmit data to another only through the SENDing of messages.

Unfortunately, excursions into the world of objects are not for the faint of heart. Terms such as classes, methods, instances, and inheritance make learning about object-oriented programming something of a hurdle. However, at the Logo 85 conference, Mark Guzdial presented a set of LCSI Logo extensions, called OOLOGO (Object Oriented Logo), which provides an opportunity to experiment with SmallTalk's paradigm.

Read All about It

Recently, I was presented with a paper by Gary Drescher describing a fresh presentation of object-oriented programming. Gary was instrumental in the creation of ObjectLogo at the now defunct Atari Cambridge Research Center. ObjectLogo has all the power of traditional Logos with the addition of objects. Gary's paper, "Object-oriented Logo" (draft of August 25, 1985), describes the form of ObjectLogo, and presents a syntax that simplifies the concept of object-programming. This "function-linked" approach is very different from (and more natural than) the "message-passing" approach taken in SmallTalk-like programming.

Here is a simple example. In ObjectLogo we can create an object called "Bill.Bird which will politely speak with us. We will first need to create it, i.e., make it a something.

```
?MAKE "Bill.Bird :SOMETHING
```

Now we can tell it to have a name.

```
?TELL "Bill.Bird [HAVE "Name "Bill.Bird]
```

Next, we can tell "Bill.Bird to learn a procedure called Say. Note the resemblance to Logo's DEFINE.

```
?TELL "Bill.Bird [ HOWTO Say [ [ Msg ] [
PRINT (SE :Name "says :Msg) ] ] ]
```

If we now ask Bill.Bird to perform, we should get some response.

```
?TELL "Bill.Bird [ Say "Hi ]
Bill.Bird says Hi.
?TELL "Bill.Bird [ Say [ Hello There. ]
?Bill.Bird says Hello There.
```

Without much explanation, watch this:

```
?MAKE "Oscar.Object KINDOF "Bill.Bird
?TELL "Oscar.Object [ HOWTO Say [ [
MSG ] [ USUAL [ Say :msg ] ] [ PRINT [ Now I
say, SCRAM! ] ] ] ]
?TELL "Oscar.Object [ Say [ Hello
There. ] ]
Bill.Bird says Hello There.
Now I say, SCRAM!
```

Interesting, eh?

NLXual Challenges continued

ObjectLogo is about to be released for the rest of us who happen to own Macintoshes. Written by Coral Software Corporation, this new version of Logo promises to be the most powerful Logo yet released. But you don't have to wait to start using ObjectLogos just because you don't have a Mac.

I sent a copy of Gary's work to Mark Guzdial. Mark was so entranced with the draft of Drescher's article that he built a set of Logo extensions for users of Apple Logo //. Called Pseudo Object-Oriented Logo (POOL), it is not as fast or bug-free as a commercial implementation, but it is great fun as well as being a fantastic learning tool.

During the past week, I have taken Mark's program and modified it for Microsoft's Logo for the Macintosh, as well as for the original Apple Logo. These Logo procedures are free for the asking from the NLX.

Why would you want to get this set of procedures? Here's one reason why. Guess what this set of ObjectLogo commands will do:

```
?MAKE "TURTLE1 KINDOF "TURTLE
?MAKE "TURTLE2 KINDOF "TURTLE
?TELL "TURTLE1 [ REPEAT 4 [FD 50 RT 90] ]
?TELL "TURTLE2 [REPEAT 36 [FD 10 LT 10]]
```

Yes! Multiple turtles! I have had at least four turtles running on a 64K Apple using Apple Logo!

The Object of the Challenge

This month's challenge is simple. Mark Guzdial has agreed to make his source code available to interested Logophiles. Discover what you can do with Pseudo Object-Oriented Logo and share your experiences with us. Send a *stamped* (39¢) self-addressed envelope to:

NLXual Challenges - POOL
Attn: Robs Muir
1688 Denver Ave.
Claremont, CA 91711

We will send a program listing for Apple Logo, Apple Logo //, and MacLogo, along with some preliminary tips for using Pseudo Object-Oriented Logo.

In addition, Coral Software Corporation has agreed to provide final drafts of Gary Drescher's paper, "Object-Oriented Logo." In the paper, Gary also proposes how ObjectLogos might be used with children, including a suggested sequence of introduction. For a copy of Gary's paper, send a *stamped* (39¢) self-addressed envelope to:

Coral Software Corporation - Dresher paper
Attn: Jeremy Jones
PO Box 307
Cambridge, MA 02142

(While you're at it, you might also request some literature on their soon-to-be released ObjectLogo.)

Please send your experiences with this program so that we can continue to improve its performance. I can also be reached via CompuServe [70357,3403] or BITnet [MUIRR@CLARGRAD].

Try herding turtles, just for fun!

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

Logo Notes

MusicLogo is now available from Terrapin, Inc. This set of Logo language and programs was developed in conjunction with Jeanne Bamberger, a noted Logo music advocate and music theorist from the Massachusetts Institute of Technology. The product combines the power of Logo with the potential of the ALF synthesizer, resulting in control over six voices, each with a range of five octaves. The package includes a MusicLogo language disk and backup, a projects disk, and a 190-page sourcebook. The ALF synthesizer must be purchased separately. The package costs $99.95. Write to Terrapin, Inc., 222 Third Street, Cambridge, MA 02142, or call (617) 492-8816.

Also available from Terrapin is *The Logo Project Book: Exploring Words and Lists*, by Alison Birch. An activities disk is included. The cost is $19.95.

The National Logo Exchange, copyright © 1986 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subscription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, VA 22905. Published since September 1982. ISSN 0734-1717. CompuServe 70020,223.

Editor..Tom Lough

NLXtra

A Supplement to the National Logo Exchange Newsletter

LIFT's

Teacher to Teacher
A Review of New Logo Materials

by
Anne Cairns Federlein

Here are two excellent new Logo resources for a wide age range of children. The following books have been reviewed and tested with elementary school children.

Beginning Students

Logo Works: Lessons in Logo, and *Logo Works Teacher's Manual,* by Sheila Cory and Margie Walker, 1986, published by Terrapin, Inc., 222 Third Street, Cambridge, MA 02142.

Target Audience: Upper elementary children

Content: The authors have researched every component of this book with children and teachers in their school district. In fact, they are to be commended for taking the time to write such a well organized, concise book after teaching all day.

The book is written for use with Terrapin Logo. Each chapter is written in sequential order, with concepts building from skills needed for beginning Logo to problem solving. The seven chapter headings are Getting Started, REPEAT, Writing Procedures, Polygons, Circles, Problem Solving, and Variables & Recursion. The Appendix includes definitions, a command summary, and clear instructions on using this book with Apple Logo, Commodore Logo, and early versions of Terrapin Logo.

An activity disk is included, containing files for Terrapin, Apple, and Commodore Logo.

Strengths: The format begins with introducing upper elementary children to the turtle. Three statements in the beginning give the philosophy of the text and the organization of each chapter.

"Narratives introduce you to something new and important about Logo." Each chapter begins with a narrative explaining the new concepts to be learned.

"Activities give you practice with the new things you have learned." Each chapter gives many ideas for activities in order to provide adequate time for the children to practice the new ideas.

"Explorations inspire you to learn more about Logo and discover on your own." Once the child has practiced the new ideas in the lesson, then open-ended suggestions are given. This allows the child to apply the learning that has taken place.

Older children will find each part of the book stimulating and challenging, especially the section on exploration. Teachers, even the Logo novices, will find the book easy to understand and use as an effective teaching tool.

Intermediate and Advanced Users

Teaching and Learning with Logo, by Allan Martin, published by Teachers College Press, 1234 Amsterdam Avenue, New York, NY 10027.

Target Audience: Middle school and high school teachers

Content: This book contains chapters on the background of Logo, using Logo in the school, turtles and trucks, turtle graphics, making music with Logo, and case studies on thinking about Logo in the classroom. Additional chapters are about the floor turtle, turtle learning in the home, turtles for special children, Logo music in the classroom, and Logo in control. The final chapters are about language with Logo, learning programming with Logo, software in Logo, a case study on Scottish schools one hundred years ago (the author is an educator in Scotland), the future of Logo, and a concluding section on Logo resources.

Strengths: The case studies are wonderful! Allan Martin not only discusses the theory and methods of using Logo in the classroom, but further expands the information by applying classroom practice. All too often in education, the value of the case study is overlooked to the exclusion of theory. The case study method in America seems to be "reserved" for psychology, special education, and medical textbooks.

The chapter on the case study of the Scottish schools of the last century is important because it helps teachers understand the ecology of the classroom and the role of Logo. Subsequently, the chapter on the future addresses the frontiers, new horizons, and the human context.

Martin predicts, "By some future stage, it may well be that Logo is no longer called Logo. Its essential features may be incorporated in a computing environment which will supersede it. Recently developed computer languages like Smalltalk and Boxer enable whole classes of objects with user-defined properties to be created: They could come to offer much in the way of the characteristics of a number of current computer environments. Those

Teacher to Teacher continued

who see the value of what Logo can offer will not mourn the eventual passing of Logo if its successor makes available all that Logo can, plus more, whilst retaining the accessibility and the creative and constructive potential which Logo offers."

This book is highly recommend for all teachers. However, when one reads this text, the question that comes to mind is, "When is the future?" The recent introduction of LogoWriter by Logo Computer Systems Incorporated and Robs Muir's "NLXual Challenges" column on object oriented Logos in this issue seem to suggest that the future is now!

Please contact the Logo Information For Teachers (LIFT) group with information on your favorite new Logo materials and books. Write to LIFT, PO Box 5396, Plymouth, MI 48170.

Anne Cairns Federlein is an associate professor of child development at Oakland University, Rochester, MI, and the editor of *The Journal of the Michigan Association for Education of Young Children.*

NLX Index, Volume 4, September 1985 - May 1986

Articles:

Blitman, Elaine, "Logo Teacher Training," (guest editorial) Nov 85
Burrowes, Sharon, "Procedure Inputs: Local Variables or Not?" Apr 86
Dawson, Sandy, "Online from Paradise: World Logo Conference Report," Dec 85 NLXtra
Harris, Judi, "Euler and the Turtle," Mar 86
Humphreys, Ceil, "Logo Teaching: A Marriage of Theory and Practice," Feb 86
Hurley, James, "Multiple Turtles Without Sprites," Oct 85
Miller, May Lou, "Logo Teacher Training," (guest editorial) Nov 85
Nutting, Kathleen, et al., "Logo 85 Report," Sep 85
Robinson, Louise, "Teachable Logo Moments: Jimmy Comes to Class," Nov 85
Schroeder, Doris, "Come to the LogoShow!" Dec 85
Sharp, Parnelli, and Lovely, Gail, "West Coast Logo Conference Enjoyed by All," Jan 86 NLXtra
Stouffer, Marge, "Sometimes Logo Teachers Should Teach," Jan 86
Tempel, Michael, and Bourgoin, Mario, "IF: At First You Don't Succeed," Sep 1985 NLXtra
Tobias, Joyce, "Logo in the United States: A Survey," Sep 1985
Watt, Dan and Molly, "Logo Quilt: A Collaborative Learning Project," May 86
Watt, Molly, "Twenty Powerful Ideas," Sep 1985
--- "Imagewriter Overprinting Solved for Apple Logo II!" Feb 86 NLXtra

Regular Features:

Bull, Glen, and Cockran, Paula, *Listful Thinking*
"Anatomy of a Procedure," Jan 86
"Blending Logo and Language Arts," Sep 1985
"Instant Tools," Dec 85
"Logo Word Chemistry," Oct 85
"Story Time," Feb 86
"Summing Up," Mar 86
"Turtles and Lists: Never the Twain Shall Meet?" Apr 86
"Turtles, Lists, and Spaceships," May 86
"Words, Names, & Starmaps," Nov85

Elias, Barbara, *Logo Disserts: Dissertations Dealing with Logo*

Fry, James, et al. *Teacher to Teacher: A Review of New Logo Materials*
Bredehorn, George, *KNOW LOGO: A Logo Exercise Program*, Nov 85
Bull, Glen, et al., *Nudges: Apple Logo Projects*, Apr 86
Charischak, Ihor, *Logo Microworld: Picture Puzzles*, Nov 85
Cory, Sheila and Walker, Margie, *Logo Works: Lessons in Logo*, May 86 NLXtra
Glantz, John, *Logo Lessons, Volume I*, Mar 86
Heller, Rachelle, et al., *Logoworlds*, Jan 86
Hoglund, Barry, *Logo Aide*, Dec 85
Kwok, Annie, *The Workbook for Learning Logo*, Sep 85
Marrapodi, Maryann, *Logo for the Apple II/IIe/IIc*, Oct 85
Martin, Allan, *Teaching and Learning with Logo*, May 86 NLXtra
Martin, Donald, et al., *Apple Logo Programming Primer*, Dec 85
Martin, Kathleen, and Bearden, Donna, *Mathematics and Logo - A Turtle Trip Through Geometry*, Apr 86
Thompson, Patrick, *Seven Microworlds for the Commodore 64*, Mar 86
Tipps, Steve, et al., *Nudges: IBM Logo Projects*, Apr 86
Tobias, Joyce, et al., *Beyond Mindstorms: Teaching with IBM Logo*, Sep 85
Watt, Molly and Dan, *Teaching with Logo*, Feb 86
Yule, David, *Apple Logo for Kids*, Oct 85

Lough, Tom, *From the Editor*

McCauley, Jim, *Q and A*

Muir, Robs, *NLXual Challenges*
"The BASIC Challenge," Oct 85
"Bug Hunting," by E. Paul Goldenberg, Sep 85
"Ella and the n-gon," Nov 85
"It's a Plot!" Mar 86
"It's Only Words," Jan 86
"Logo Collectibles," Feb 86
"Making List Operators," Dec 85
"Objective Orientation," May 86
"Round and Round," Apr 86

Poplin, Rebecca, *Logo Teacher Feature*
"Featuring: Peter Anderson," Feb 86
"Featuring: Jim Fry," Apr 86
"Featuring: Judith Harris," Jan 86
"Featuring: Mark Mariotti," Mar 86
"Featuring: Robs Muir," Sep 85
"Featuring: Peter Rawitsch," Dec 85
"Featuring: Sebastian and Sandra Reisch," Nov 85
"Featuring: Leslie Thyberg," Oct 85
"Featuring: Sue Wolff," May 86

Tipps, Steve, *Tipps for Teachers*
"Circular Quirks," Mar 86
"Community Builders," Nov 1985
"Dash Away All," Sep 85
"Geologoboards," Dec 85
"Getting to Know You," Feb 86
"Goaltending, Year's Ending," May 86
"Overland Mapmaking Adventures," Jan 86
"Random Redux," Apr 86
"Rooms for Logo," Oct 85

Toth, Jane, and Lanyi, Donna, *TurtleTips*

Wigley, Griff, *NLXionary, A Lectionary of Selected Logo Readings*