

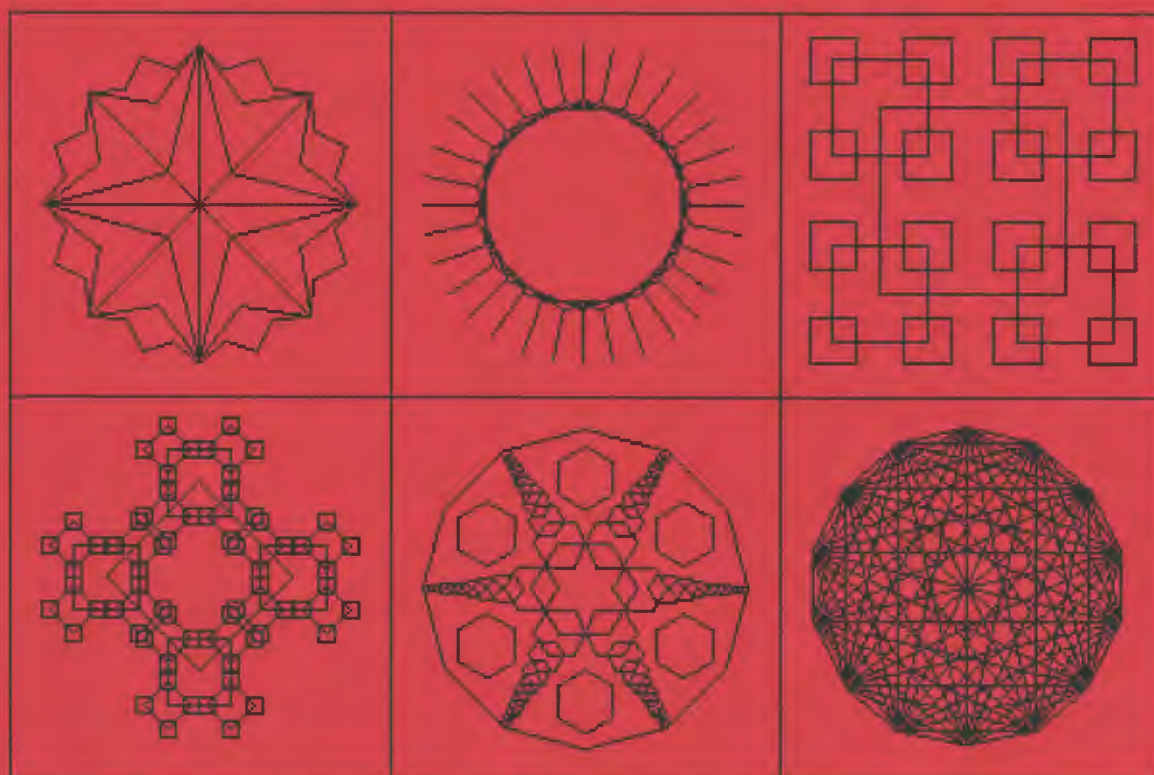


# LOGO EXCHANGE

Volume 14 Number 2

Winter 1995-96

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



Variations on a Theme, from "Turtle on Tour"



***In this issue:***

**Learning With a Ball of String**

**From "Uh-huh" to "Ah-hah"**

**The Wonders of String Art**

**Logo Fan and Thermostat**

**Does Logo Learning Increase Over Time?**

**Countdown...With Robotics!**

**Commercials—Logo Style**

**Turtle on Tour**

**High School Geometry**

**Editorial Publisher**

International Society for Technology in Education

**Editor-in-Chief**

Dorothy M. Pritch

**Copy Editor**

Jennifer Lindsey

**Founding Editor**

Tom Lough

**International Editor**

Dennis Harper

**SIGLogo Officers**

Chuck Friesen, President  
Steve Sesko, Vice-President  
Hope Chaffin, Secretary/Treasurer  
Dorothy Fitch, Editor

**Production**

Kerry Lutz

**Editorial Coordinator**

Jodie Rogers

**Director of Advertising Services**

Lynda Ferguson

**Marketing Director**

Martin G. Boyesen

**Submission of Manuscripts**

*Logo Exchange* is published quarterly by the International Society for Technology in Education, Special Interest Group for Logo-Using Educators. *Logo Exchange* solicits articles on all aspects of Logo use in education.

Manuscripts should be sent by surface mail on a 3.5" disk (where possible). Preferred format is Microsoft Word for the Macintosh. ASCII files in either Macintosh or DOS format are also welcome. Submissions may be made by electronic mail as well. Where possible, graphics should also be submitted electronically. Please include electronic copy, either on disk (preferred) or by electronic mail, with any paper submissions. Paper submissions may be submitted for review if electronic copies are supplied upon acceptance.

**Send surface mail to:**

Dorothy M. Pritch  
1 Derby Road  
Derry, NH 03038

**Send electronic mail to:**

Internet: 71760.366@compuserve.com

**Deadlines**

To be considered for publication, manuscripts must be received by the dates indicated below.

Volume 14, Number 1	Jan. 1, 1996
Volume 15, Number 1	Mar. 1, 1996
Volume 15, Number 2	June 1, 1996
Volume 15, Number 3	Oct. 1, 1996

**ISTE Board of Directors 1995-96**

David Brittain, President  
MGT. of America (FL)  
David Moursund, Executive Officer  
Dennis L. Bybee, Associate Executive Officer

**Executive Board Members**

Connie Stout, Secretary  
Texas Education Network (TX)  
Barry Pitsch, Treasurer  
Hartland AEA RTI (IA)  
M.C. (Peggy) Kelly, Past-President  
California State University - San Marcos  
Terrie Gray, At Large  
Chico Unified School District (CA)

**Board Members**

Kathy Edwards  
Apple Computer, Inc., Santa Monica (CA)  
Terry Killian  
Jefferson County RT School District (CO)  
Paul O'Driscoll  
Salem-Kitter Public Schools (OR)  
Lynne Schrum  
University of Georgia  
Celia Schutte  
Educational Technology Consultant (FL)  
Gwen Solomon  
U.S. Department of Education (DC)  
Neal Stradler  
University of Nevada - Las Vegas  
Harriet Taylor  
Louisiana State University  
Breida Williams  
West Virginia Department of Education

*Logo Exchange* is published quarterly by the International Society for Technology in Education (ISTE), 1787 Agate Street, Eugene, OR 97403-1923, USA; 800/336-5191. This publication was produced using Aldus PageMaker®.

ISTE Members may join SIG/Logo for \$24.00. Dues include a subscription to *Logo Exchange*. Add \$10 for mailing outside the USA. Send membership dues to ISTE. Add \$4.00 for processing if payment does not accompany your dues. VISA, Mastercard, and Discover accepted.

Advertising space in *Logo Exchange* is limited. Please contact ISTE's director of advertising services for space availability and details.

*Logo Exchange* solicits articles on all topics of interest to Logo-using educators. Submission guidelines can be obtained by contacting the editor. Opinions expressed in this publication are those of the authors and do not necessarily represent or reflect the official policy of ISTE.

© 1995 ISTE. All articles are copyright of ISTE unless otherwise specified. Reprint permission for nonprofit educational use can be obtained for a nominal charge through the Copyright Clearance Center, 27 Congress St., Salem, MA 01970; 508/744-3350; FAX 508/741-2318. ISTE members may apply directly to the ISTE office for free reprint permission.

**POSTMASTER:** Send address changes to *Logo Exchange*, ISTE, 480 Charnelton St., Eugene, OR 97401-2626 USA. Second-class postage paid at Eugene, OR. USPS# 650-130. ISTE is a nonprofit organization with its main offices housed at the University of Oregon. ISSN# 0888-6970.

## Contents

From the Editor: Try Something New! .....	<i>Dorothy M. Fitch</i>	2
Quarterly Quantum: In Praise of Apollo 13 .....	<i>Tom Lough</i>	3
Learning With a Ball of String .....	<i>Jim Muller</i>	5
Creating a Commercial—Logo Style .....	<i>Diane M. Manuel</i>	8
From “Uh-huh” to “Ah-hah”: Logo in Action .....	<i>Robert M. Berkman</i>	10
Countdown...With Robotics! .....	<i>Manorama Talaiver</i>	13
LEGO Lunar Landers .....	<i>Michael Lipinski</i>	17
The Wonders of String Art .....	<i>Jim Muller</i>	18
Turtle on Tour .....	<i>Morton Goldberg</i>	20
High School Geometry Using Logo .....	<i>Robert M. Packard</i>	27
Control Simulations (Part 2) Fan and Thermostat .....	<i>Glen L. Bull, Gina L. Bull, and Todd Kent</i>	29
Logo: Search and Research Does Logo Learning Increase Over Time? .....	<i>Douglas H. Clements and Julie Sarama</i>	34
Global Logo Happenings .....	<i>edited by Dennis Harper</i>	
The Use of Logo With Disadvantaged Children .....	<i>José Armando Valente</i>	37

# From the Editor: Try Something New!

by Dorothy M. Fitch

I have just finished reading a book for the third time, which puts it in the same league as Seymour Papert's *Mindstorms*. In fact, I thought a lot about *Mindstorms* as I read it, even though this book is neither about Logo nor about computers. It is, however, about learning.

You may be familiar with John Holt, who has written many books about learning, such as *How Children Fail*, *How Children Learn*, and *Learning All the Time*. His book *Never Too Late* is also about learning, but more about how adults learn and how they often learn differently from children.

*Never Too Late* is a fascinating account of how, at the age of 40, Holt took up the cello and how he studied his own learning. Although this book may be of particular interest to those who have studied a musical instrument, there is much to learn about learning here. This book especially spoke to me as I too, at 40, took up the cello.

By the time I finished the book (the first time through), I desperately wanted to talk to Holt about Logo and introduce him to Seymour Papert. (Unfortunately, Holt passed away in 1985, though this book is a recent release of a 1978 publication.) Did John Holt and Seymour Papert ever meet? Did Holt know about Logo? I'd be curious to know....

In any event, Holt's points and analogies are exciting to think about in relation to Logo.

— People can build structures of knowledge that are far more powerful than the specific pieces of information they encompass. Rather than list how many ways one can get from point A to point B on a map, we can internally build the structure of the map, which allows us to unconsciously use all the information that map can impart. Structures of knowledge are greater than the sum of their parts.

— How do adults learn differently from children? Only adults can invent problems and then find solutions to them. Aware of our own deficiencies, we can devise ways of helping ourselves improve.

— On the importance of learning how to learn: "Part of the art of learning any difficult act, like

music, is knowing both how to teach yourself and how best to use the teaching of others, how to gain from the greater experience and skill of other people without becoming dependent on them. For few people are likely to become good at music, or anything else, who do not learn how to teach themselves. What we can best learn from good teachers is *how* to teach ourselves better."

— A parent at a school where Holt taught complained about the excessive testing: "You're like a gardener who constantly pulls plants up by the roots to see if they are growing." We shouldn't have to ask daily, "Am I learning anything? Am I learning enough?" Holt says, "If we're working, doing our best, challenging ourselves, doing things that are hard for us, and sometimes very hard, paying attention to what we hear in our ears and feel in our muscles, not letting our minds wander or think about something else—if we are fully involved in our music making, interested in it, excited by it, then we are learning." Substitute any activity for music making.

Holt points out that if we truly want to do something, we can make the time for it, although there will be choices and sacrifices.

As you start the new year, find the time—*make the time*—to learn something new. Take up tennis, birding, skiing, pottery, cello. You will be reinvigorated, and your enthusiasm will benefit your students. You will remember what it is like to learn something new, and you can study how *you* learn. ▲

**Dorothy M. Fitch**, LX Editor  
3 Derby Road, Derry, NH 03038  
603/425-2010  
E-mail: 71760.366@compuserve.com

## References

- Holt, J. (1991). *Never too late*. Reading, MA: Addison-Wesley.
- Papert, S. (1980). *Mindstorms: Children, computers and powerful ideas*. New York: Basic Books, Inc.

# Quarterly Quantum: In Praise of Apollo 13

by Tom Lough

---

By the time you read this, I hope you will have already seen *Apollo 13* (starring Tom Hanks) at least once. What a film!

Naturally, I found myself thinking about Logo as the saga unfolded. Shortly after launch, for example, it was fun to think “in Logo” while watching Jack Swigert (played by Kevin Bacon) maneuver the Command Module to dock with the Lunar Module (LM).

```
to dock
forward 10
forward 10
forward 10
right 180
setheading towards :lm
forward 10
forward 10
forward 2
etc.
```

The explosion that crippled the mission occurred about a day after the docking maneuver. While observing the responses to this critical situation, I suddenly became aware of more major educational aspects of the film.

After making the decision to use the moon as a “slingshot” to return the spacecraft to earth, NASA Mission Control had to change the orbit of Apollo 13 accordingly. For me, the moment of the rocket burn to make the orbital changes trumpeted the critical importance of a solid background in science, mathematics, and computer technology. The members of Mission Control “knew their stuff,” and were able to complete the required calculations accurately and in time.

I marveled at the way in which everyone worked together in an extremely stressful situation. The lives of three men were at stake. The magnificent commitment to teamwork shown in this film should serve as a model for students (and teachers) everywhere.

The impact of effective leadership and a positive attitude was striking. Several critical points in the film turned on the leadership and positive attitude manifested by Gene Kranz, the flight director, played by Ed Harris. Such an attitude, when coupled with an unfailing persistence, rarely yields to failure. What a message for students.

Both the Apollo crew and Mission Control demonstrated their creative abilities as they solved practically every problem that could go wrong, and then some. The classic moment was figuring out how to use a square carbon dioxide filter in place of a round one. We joke about fitting a square peg into a round hole, but this actually happened! The ingenious and unconventional solution was not in anyone’s “answer book,” but it did the job.

The respect for learning and the uncommon dedication to working together to apply collective knowledge in creative ways spoke strongly to me. I began this column by commenting on the obvious connections to Logo that the movie suggested. But I believe that the magnificent accomplishments of the Apollo crew and the Mission Control team in this episode from real life contain some of the most worthwhile educational lessons we could ever wish for our students.

I believe our challenge is in finding ways to capitalize on these lessons. I also believe that we have a fantastic tool in Logo and Logo-based applications.

For learning about science, mathematics, and computer technology, Logo has a lot to offer. For fostering cooperative learning and teamwork, Logo has few peers. Problem solving is one of the main attractions of Logo-based lessons. Creativity, persistence, positive attitudes—all are found in Logo learning environments. How can we put it together for our students?

At the next opportunity, go to see *Apollo 13* again. Be “educationally alert.” Watch for the



science, the mathematics, and the computer technology as the drama plays out. Appreciate once again the fantastic teamwork and the incredibly creative problem solving that took place. Observe the critical contributions made by positive attitudes.

What a challenge to bring the educational flavor of that film into our classrooms. We have our Logo knowledge and experience. How can we help our students realize that one small step for a turtle could lead to a giant leap for all mankind? ▲

**FD 13!**

**Tom Lough**

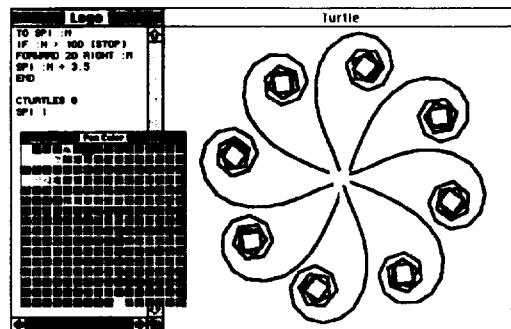
Founding Editor

Box 394

Simsbury, CT 06070

E-mail: 70020.223@compuserve.com

# How Can Logo Help You Teach?



*New Logo PLUS for the Macintosh lets your students explore—in glorious color!*

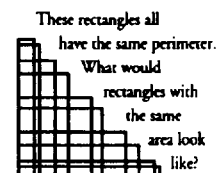


Logo is a teaching tool unlike any other. It is active and hands-on. Kids can explore, making their own discoveries about angles and measurements, patterns and shapes. With Logo, rotations, reflections, translations, and tessellations are a snap.

Students estimate, plan, and create, using a variety of problem-solving and debugging strategies. Students can figure out on their own how to draw arcs, circles and polygons, or they can use the handy built-in tools. They can explore fractions, use variables, plot points, graph functions. In short, your students can explore your entire math curriculum using Logo.

Logo teaches process, the "how-to" of math. It says "explore and see for yourself." And in learning step-by-step, students acquire the sort of knowledge that sends them confidently in many new directions.

With over 50 tools for exploring all areas of math, Logo PLUS is the natural and logical choice of a language for your classroom.



Call or write for our free *Exploring Math with Logo* and *Why Use Logo?* booklets.

**1-800-972-8200**

***Terrapin™***

10 HOLYWORTHY STREET • CAMBRIDGE, MA 02138  
617-547-LOGO (547-5646) FAX 617-492-4610

*Logo languages and curriculum materials for all machines.*

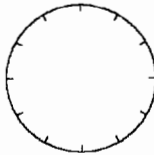
# Learning With a Ball of String

by Jim Muller, CIS Logo Forum

---

When my co-authors and I were developing *The Turtle's Sourcebook* in the early 1980s, one of the most effective off-computer activities we found was what you could do with a ball of string. No single tool helped Logo come more alive to the young people with whom we worked.

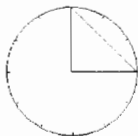
We started with a circle drawn on the pavement outside. One "turtle" stood at Home in the center of the circle holding the end of a ten-foot (or thereabouts) length of string. Another turtle wrapped the end of the string around a piece of chalk. Stretching the string taut, the second turtle drew a perfect circle around the center turtle. It was then fairly easy to mark off the hours of the clock on the chalk circle.



When the turtle faces 12:00 o'clock on the circle it is the same as the turtle facing the top of the screen. It does not take much imagination to see that the clock circle can lead to all sorts of interactive and physical practice with angles and degrees.

But what about the string?

Make a game out of it. Position one child at each hour and another at Home with a ball of string. The string becomes the lines that the turtle draws. Toss the string from Home to 12:00 o'clock. That child turns and tosses the string to 3:00 o'clock. That child then tosses the string Home. What type of shape did the string create?



What happens when the children at 12:00 o'clock and 3:00 o'clock hand their corners of the string triangle to the children standing at 1:00 o'clock and 4:00 o'clock? In effect, you are moving a triangle through space, a rather simple demonstration of a complex process.



There's another way to use this same concept, but on an individual basis.

You are going to need a piece of wood from which you can cut a square that is about 12 inches on each side. Use a piece of 12-inch shelving or a piece of plywood. You are also going to need at least thirteen small nails about 1½-inches long, a hammer, a pencil, and about 12 feet of yarn. String will do.

1. Hammer a nail into the center of the board ... just part way so that you have an inch or more sticking up from the board.
2. Tie the string to the nail in the center. Now you are going to make a circle around the center nail.
3. Stretch the string out to the edge of the board.
4. Put your pencil out near the edge of the board. (Hold it up straight.) Then wrap the string around the pencil.
5. Hold the pencil up straight and stretch the string out from the nail. Then draw a circle around the center nail.

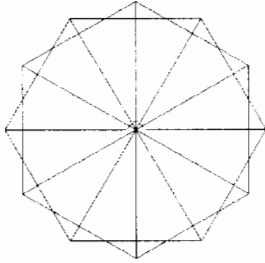
Now you are ready to hammer the other 12 nails into the board. But first, do you have a printer that can print pictures? If so, the turtle can make you a pattern for your string board.

Start with this procedure to draw a triangle with equal sides.

```
to tri :n
repeat 3 [forward :n right 120]
end
```

Now use that procedure to make a pattern. You can make the :n variable as large as required.

```
repeat 12 [tri 100 right 30]
```



The result is a pattern with 12 points, just like the numbers on a clock. Print the screen.

Now carefully push the pattern over the nail in the center of the board. Hold or tape the pattern in place. Then draw a line along each of the pattern lines to the circle you drew before. Where each line crosses the circle, hammer in a nail.

Now each child can play turtle graphics using a ball of string and their very own string board.

Take the string from the center nail (that's Home on the screen) and stretch it up to the nail at 12:00 o'clock. Go around the 12:00 o'clock nail and take the string around the 3:00 o'clock nail. Then take the string Home to the center.

Now do the same thing, only go around the 1:00 o'clock nail and the 4:00 o'clock nail. Next, go around the 2:00 o'clock nail and the 5:00 o'clock nail, the 3:00 o'clock nail and the 6:00 o'clock nail, the 4:00 o'clock nail and the 7:00 o'clock nail. Continue all the way around.

What pattern has the string made? It looks just like the one you printed to create the string board.

Your hand acted like the turtle as it moved around the string board. But instead of drawing lines on the screen, you made a line of string.

Take a close look at this drawing. We have been talking about using triangles to create this

pattern. But do you see some other shapes here? What kind? Where?

Using a ball of string focuses attention on the planning required to create a graphic. The players need to develop a plan and visualize the steps. No one particularly likes to rewind many feet of string after making a mistake.

### Some other individual activities

The first thing you will need is a shoebox. Paint the inside of the top using flat black paint. This creates a dull background to show off your string patterns.

The next job is to create an even pattern that you will use to punch tiny holes evenly around the edge of the box top. You can do this very easily on the computer

```
to pattern :dist :marks
if :marks = 0 [stop]
forward :dist
mark :dist
pattern :dist :marks - 1
end
```

```
to mark :dist
right 90
forward :dist / 10
back :dist / 5
forward :dist / 10
left 90
end
```

The variables let you set the number of marks (:marks) and the distance (:dist) between them. For example, if you want to print 20 horizontal marks that are 25 turtle steps apart, type **pattern 25 20** and press Enter.

Print the patterns and cut them into narrow strips. Then tape them to the edge of your painted box top.



## Curves from straight lines

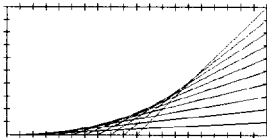
Now you can use colored yarn and a needle to make curves from straight lines.



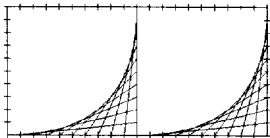
Here's a box top pattern.

1. Start at the lower left-hand corner.
2. Push the needle through the corner mark into the box top and then out through the mark at the lower right-hand corner.
3. Move up to the first mark up the right side and push the needle from the outside into the box top.
4. Go to the first mark in from the left corner and push the needle from the inside to the outside of the box top.

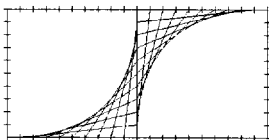
Soon you will have a pattern that looks like this, a curve made from straight lines.



There are lots of other patterns you can make. Why not try these? Of course, you must add another pattern across the middle of the box top to make these patterns.



Turn one of those upside down and look what you'll get.



Try inventing other patterns. If you want to dress them up a bit, try different colors of yarn for different parts of the design.

There is much that can be said about the value of applying Logo concepts and turtle graphics on and off the computer. A quotation of unknown origin sums it up very nicely.

"He who has imagination without learning, has wings but no feet."

Keep this in mind as you explore the wonders of a ball of string with your class.

These activities and many others are described in *The Turtle's Discovery Book*. For information, contact me at the e-mail address below. Enter GO LOGOFORUM to explore the Logo Forum on CompuServe. ▲

**Jim Muller** has had a lifelong interest in translating various technologies into understandable and persuasive programs. In 1981, Muller and his son organized the first Logo users group, the Young Peoples' Logo Association, which eventually grew into a worldwide 6,000-member organization. Among other things, the YPLA published *1,2,3 My Computer and Me* and *The Turtle's SourceBook*. Both of these books are now being expanded for use with current versions of Logo. In 1985, the YPLA merged with CompuServe where it became The Logo Forum. Today, Muller is a computer training and marketing consultant in the Dallas/Fort Worth metroplex. You can reach him via e-mail at 76703.3005@compuserve.com or on CompuServe at 76703,3005.

# Creating a Commercial—Logo Style

by Diane M. Manuel

At North Collins Elementary School, we have done many interesting Logo projects over the years. My favorite, however, is the following fifth-grade Commercial Project.

## The setting

Classes of 20 or more students come to me in the Computer Lab for 45 minutes once a week. Primary classes may have up to 24 students. Last year I had 21 Apple IIe and 2 Mac LC computers in the Lab. The principal has made a commitment to have enough computers so that each student (especially in the upper grades) can use a computer alone. This level of support is important to any program.

I am currently using *LogoWriter* in my K-6 elementary school. Although I have used other versions, such as a *Terrapin Logo* version dating back to the early 1980s, I find *LogoWriter* to be very versatile and easy to use. Because of its multiple facets, I can use it for second- and third-grade word processing as well as for introducing turtle graphics. Fourth-grade students work with polygons and build a house using a square and a triangle. Fifth graders enjoy the Commercial Project.

## The Commercial Project

To give the students a starting point for the Commercial Project, we talk about commercials in general and outline some of the properties of commercials: sound, movement, color, shape, and message.

Our Logo commercial has a “star” (**shape**), which has a **color** and **moves** in its environment, the **background**. **Music** enhances the commercial, and a message (**text**) finishes it off. Each of these features forms a module for the commercial.

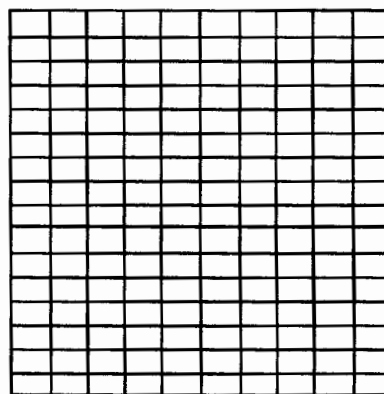
At the beginning of the project each student thinks generally about a product that he or she would like to sell. Our students have advertised pools, dresses and other clothing, farm machines, and animal products.

As we begin each of the modules, I demonstrate the commands that will be new to them. Since they have used *LogoWriter* in fourth grade for the Polygon project, they are already familiar with many of the commands.

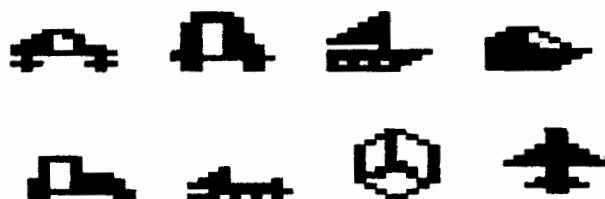
Here is the outline that I use:

**Shape**—First the students use paper grids to design their shape. Then they move to the computer to make the shapes. Each student usually creates three or more shapes. Some are very intricate, and each time I do this project I find students making more creative shapes.

SETSH \_\_\_\_



Shape grid



Examples of student shapes

**Color**—The students review the color commands. For easy reference, I hang large numbers from the classroom lights. Each number is cut from paper of the appropriate color.

**Movement (move)**—Now the shape has to move. We discuss direction and the effects of **penup** and **pendown**. Together we write a movement pattern. We review procedures and write the movement of the shape in a procedure called **move**. I stress short procedure names. Many students go off on their own and create unusual movement patterns. Sometimes they design products that move back and forth, like a floor polisher. A puppy may walk to a food dish, or a skier may come zipping down a hill. This is also the module in which they include color commands. This part of the project is the most difficult to illustrate on paper.

**Sound (music)**—We often discuss the difference between sound, noise, and music. They choose a familiar melody and, using a conversion chart, they write the music procedure. Often a student will try to use music from band or music lessons. This takes quite a bit of time. The success of this segment does not depend on a student having a background in music. Some students will finish this part of the commercial in one class period. Others will struggle with it, some only finishing one line of the song. However much they complete, they are generally pleased with the result.

**Background (bkgr)**—Because of time constraints, the background has to be very simple. It might be a store front, or the horizon line or lines to indicate a room.

**Message (text)**—Using **print [ ]** and **tab**, students learn to place text attractively on the screen. Many will use this as an occasion to write short poetry or use a play on words. This module is the last procedure and some students aren't able to complete this section because of lack of time.

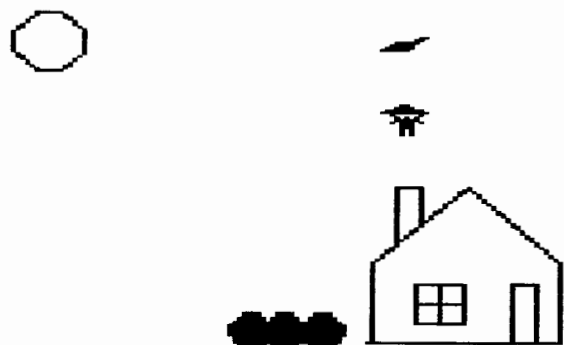
Finally, we write a procedure to run the entire commercial:

```
to c
move
music
bkgr
text
end
```

Normally, I don't like to use the same lesson more than a few times. However, I truly like this one, and the students seem to enjoy it as well. It gives them the opportunity to use many commands and put them into a project that they enjoy. ▲

**Diane M. Manuel** is the Computer Education teacher for grades K-6 at North Collins Elementary School in New York. She has been involved with various forms of Logo since the early 1980s and has provided inservice training for teachers on computer-related subjects, including *LogoWriter*. She is also the Gifted and Talented catalyst for the elementary school. Diane can often be found in the "Electronic Schoolhouse" on America Online. She can be reached at North Collins Elementary School, 10469 Bantle Road, North Collins, NY 14111 or via e-mail at [DianeTeach@aol.com](mailto:DianeTeach@aol.com).

### Samples of student work



*In this commercial, the plane flies from the right and drops the parachuter, who falls into the chimney, as planned!*

Buy our new flying TRAINS.  
CALL 337 - 3940



# From “Uh-huh” to “Ah-hah”: Logo in Action

by Robert M. Berkman

Life can be difficult for a Logo believer, especially if you work in an environment where allies are few and far between. Although my school is rich in computer gear, its capabilities are rarely stretched beyond word processing. To me, this is the equivalent of ordering \$50,000 worth of paints and brushes, then stipulating that students only be allowed to paint landscapes. Landscapes are wonderful things, but if we really want to use these materials to their full potential, we should encourage children to paint portraits, abstracts, storyboards, optical illusions, and even parabolas. Unfortunately, confining computers to word processing tasks happens all too frequently in schools.

Thus, when my students encounter Logo for the first time at the ripe old age of 13, it is somewhat of a revelation. Many have an image of computer programming as somewhat complicated or esoteric. Their encounters have been with games, paint programs, or word processors—programs that have been written *for* them but not *by* them.

As a concerned teacher, I want to make sure that their first encounters with Logo are positive, yet I also know that a certain amount of prerequisite knowledge is necessary to make anything interesting happen. Thus, my students have to have a set of “uh-huh” experiences before they can reach an “ah-hah.”

To introduce my students to some basics of programming (while sneaking in some algebra), I have them complete a set of problems by hand, which include the following:

$3 + 2 \times 4$   
 $2 \times 4 + 3$   
 $4 \times 5 + 6 \times 2$   
 $6 \times 2 + 4 \times 5$

I explain to my students that they will use a computer to check their homework, and that all computers are programmed by mathematicians

to operate the same way. Their job is to figure out what rules these mathematicians have agreed to, based on the results of these calculations.

This allows me to introduce a small dose of computer code. First we explore operations: the fact that the `*` sign means multiplication and the `/` sign means division, where the number before the symbol is the dividend and the number after it is the divisor.

My students’ reactions to this activity begin with frustration—not with the syntax of the language (there are very few complaints about typing **print** before an expression), but with the results that emerge. Because most students have been taught to compute from left to right, their reaction is one of bewilderment when they find out that  $6 \times 2 + 4$  is equal to  $4 + 6 \times 2$ ,  $4 + 2 \times 6$ , and  $2 \times 6 + 4$ . Frustration turns to insight, however, when these young investigators realize that the computer is selecting certain operations over others. What emerges is something that my students and I will later refer to as the “o o’ os,” or the order of operations.

As a follow-up activity, the students learn to write their first procedures, in which they write algebraic expressions using local variables. As an example, I show students the following procedure, which could be used to calculate the cost of buying compact discs:

```
to cd :amount
  print :amount * 11.95
end
```

I explain that when one types a word preceded by a colon, we are defining a variable, a type of container that can hold a number. We run the procedure several times with different numbers of CDs to show how the result will be displayed.

I then give the students several ideas for procedures to write on their own. One could be

used in a fast food restaurant to calculate the bill for a certain number of hamburgers, french fries, and soft drinks. Their procedures look something like this:

```
to bill :hamburgers :fries :drinks
print :hamburgers * .99 + :fries * .89 +
:drinks * .69
end
```

Some students decide to do something more elaborate and ask how they can insert little messages into their programs, like this:

```
to bill :hamburgers :fries :drinks
print (sentence [Total for hamburgers =
$] :hamburgers * .99)
print (sentence [Total for french fries
= $] :fries * .89)
print (sentence [Total for drinks = $]
:drinks * .69)
print (sentence [Total meal = $] :ham-
burgers * .99 + :fries * .89 + :drinks
* .69)
print [Have a good day.]
end
```

As my students work on these procedures, they must tie together their understanding of order of operations, variables, and expressions. As an introduction to programming, this is an ideal activity. It gives students insight into how programs are written for restaurants (computerized cash registers), banks (interest rates), clothing stores (discounts and sales tax), and so on. Since these procedures consist of relatively simple syntax and use few primitives, students gain a lot of learning from just a little Logo.

My students now want to make their programs more interactive; they would like to ask the user some questions and then respond appropriately. I explain the difference between local and global variables, as well as the concept of a list. There is also the matter of how to take a response and place it into a variable.

```
to figureit
print [Enter number of hamburgers.]
make "burgers readlist
print (sentence [Please pay $] (first
:burgers) * .99)
end
```

```
to price
print [Enter number of hamburgers.]
make "burgers first readlist
print (sentence [You owe me $] :burgers
* .99)
end
```

One interesting discovery took place when the students realized that parentheses would be needed in the **figureit** procedure around **first :burgers**, but not needed in the **price** procedure when the global variable is defined. My students realized that computer syntax is not a lot different from the order of operations: some things must be taken care of before other things, and we can use parentheses to specify these events.

As this learning takes place, my students cycle between the “uh-huhs,” which consist of my instructing them in the language and syntax to make things happen in Logo, and the “ah-hahs” that come from their writing a program that does what they have in mind. Interestingly, the “ah-hah” experiences lead to students demanding more “uh-huhs” (instruction in accomplishing a certain task in Logo), which, in turn, create more “ah-hahs.”

For example, one thing that drove many students crazy was the space between the dollar sign and the result of a calculation:

```
Your total bill is $ 1.21
```

This spacing seemed rather peculiar to them, and removing the space became an obsession. This simple example showed my solution, which uses the primitive **word**:

```
print word "$ 5
```

This “uh-huh” quickly turned to an “ah-hah” when my students could display their perfectly formatted results. This watershed event helped my students see the power of a computer language like Logo to make anything happen. They could teach the computer to read the input of an operator, use it to run a procedure, then reply with a programmed message.

There is more to these projects than just mastering a computer language. With Logo, students can experience the power of algebra in a

context that is interesting and meaningful. Of course, these activities could be done with a spreadsheet or with another programming language, but only in Logo can a student get a lot of "ah-hahs" with a minimum investment in "uh-huhs." Students can write complex procedures without memorizing a lot of code; yet at the same time, there are enough commands for students to make many different things happen while applying the algebra concepts they are studying.

As I stated in the beginning of this article, a Logo believer's life can be a lonely one. However, this does not mean that we have to abandon our mission. There is plenty of room for Logo in the curriculum. It doesn't have to be "taught" in a stand-alone computer class, and it can enhance the relevance of a topic like algebra. Our struggles may be in solitude, but persistence proves that they are never in vain. ▲

**Robert Berkman** is a math/science teacher at the Bank Street School for Children, 610 West 112th Street, New York, NY 10025. He can be reached via e-mail at [RBerkman@Bnk1.bnkst.edu](mailto:RBerkman@Bnk1.bnkst.edu) or at [Quickbyte@aol.com](mailto:Quickbyte@aol.com)

## New Address for Terrapin Software

Terrapin Software, Inc., developer of Logo language programs and publisher of Logo-based curriculum materials, announces a change of location. The new address, as of November 15, 1995, is:

Terrapin Software, Inc.  
10 Holyworthy Street  
Cambridge, MA 02138  
tel: 800-972-8200  
tel: 617-547-LOGO (547-5646)  
fax: 617-492-4610  
e-mail: [71760.366@compuserve.com](mailto:71760.366@compuserve.com)

At its new location, Terrapin will continue to strengthen and expand its offerings through cooperative marketing and joint product development with Harvard Associates, Inc., another major developer of Logo products

Please contact Terrapin at its new address to receive a catalog and free subscription to its new series of newsletters.

# Countdown...With Robotics!

by Manorama Talaiver

---

[Editor's note: This article and "LEGO Lunar Landers", on page 17, were originally published in the April 1991 issue (Volume 2, Number 3) of LEGO DACTA CONNEXION™, a newsletter for *LEGO® TC logo* Users. Reprinted with permission of LEGO Dacta, the educational division of LEGO Systems, Inc. All the products referenced in this article are still available from LEGO Dacta.]

Are you looking for an exciting project for your class? Are you searching for ways to integrate science, mathematics, and other curricular areas? Do you need a fun way to excite your students in a summer program? Read on about Space Station 2000.

Since 1984, the Mathematics and Science Center in Richmond, VA, has offered space education as one of two strands in its summer school program. When it was time to plan the 1990 curriculum, I chose *LEGO TC logo* as the material students used for modeling machines for future human settlements in space. Inspired by NASA's Space Station Freedom (a program to develop an inhabited laboratory in space), I called our program Space Station 2000.

The Space Station 2000 course was offered to a total of 84 gifted fifth- and sixth-grade students in two separate week-long sessions at the Governor's Summer School held at our center. The program was cosponsored by the State Department of Education.

Our general goal was to integrate space science, physical science, and robotics concepts with Logo programming. We developed four instructional modules: Space Station, Simple Machines, Robotics, and Logo. In addition to science and math concepts, cooperative learning was an important part of our approach, so we organized our materials for student groups of four. (See the end of the article for a list of materials.) Five teachers from science, robotics, and computer curriculum areas were involved in instructing and helping students. Classes were held from 8:30–3:30 for five consecutive days. Before classes started, we held an orientation session.

## Orientation

Parents and students attended the orientation session, which was designed to acquaint students with information about space and set the stage for the summer school activities, especially designing *LEGO TC logo* models as the functioning components of the space station modules. After receiving a package of building and reading materials, students were asked to build a space station and answer some questions from their reading.

Each student designed his or her own space station module. Some students designed research stations for biological or geological research; others made recreation and food production modules.

## Day 1: Astronaut training

On the first day, students participated in astronaut training to learn more about the space station concept. In space, astronauts must perform their tasks correctly and work well together, so communication skills and ability to follow directions are important. We helped focus students on these skills by asking them to give verbal directions for building a LEGO model. Here was how the activity worked.

One student built a model and then hid it. To his or her partner, the student gave verbal instructions for putting each LEGO element together to form the model. After the partner built the model from verbal instructions, all students analyzed and discussed the similarities and differences between the original model and the copy. Through this discussion students understood why differences occurred and how their communication skills could change to improve the result.



After the communication activity, students learned *LEGO TC logo* commands for controlling LEGO lights: **talkto**, **on**, **off**, **onfor**, and **flash**. Then the groups designed and programmed a lighting system for the space station cargo bay. The only criteria set for this problem-solving activity were that each group must use four lights and two of the lights must be red and flash at all times.

## Day 2: Generating interest

This was the day for experimenting with physical science concepts. After a lecture in which an egg beater and bicycle gears were demonstrated, students with Technic II sets built the "spaghetti machine" (on the back page of card number 2) to see how gears work.

Then students built other devices to see how different gearing systems could be designed and to understand the concept of speed versus power using various gear ratios. One especially exciting moment involved a pair of students who asked me if I would give them a belt, motor, light brick, and a connecting wire. By changing the gears, they created a generator to light the brick without connecting it to a battery pack or to the computer! [Note: See page 47 of the *Teacher's Guide to Technic II* for information on using the motor as a generator.] Once a generator system was created, every group was motivated to build unique lighting using generators to power them. In addition to the LEGO Dacta materials, students used Optical Data's *Living Textbook: Physical Science* videodisc as a reference for pulleys, gears, and simple machines.

In the afternoon, students were introduced to **listento** and **tone** commands. Then we asked the students to build a device for monitoring rats at the biological research module. The rat had to be kept in a special environment. If anyone touched the rat's "home," an alarm had to sound. Children used either the touch sensor or the optosensor to program an alarm system. For example, students can use the following procedure:

```
to alarm
  listento 6
  waituntil [sensor?]
  repeat 5 [tone 440 30 wait 1
    tone 880 1 wait 1]
end
```

At the end of the day, students discussed the devices they wanted to build for their personal projects. In the remaining days of the course, students concentrated their energies on their own inventions.

## Day 3: Communication and recreation

On the third day, students spoke with two educators via telecommunications to discuss invention proposals. Some students designed and programmed recreational devices for a family with two children living in the space station.

One especially creative project was the Video Communication Module. An astronaut was shown relaxing in a room watching television. A touch sensor was programmed to open the cabinet where the "television" was stored. (The "television" was the white window element in the Technic Control 0 set.) Pressing another touch sensor turned on the television (a LEGO light turned on). A security system monitored the area outside the television room for "aliens" who might be "plotting to attack" the astronaut. The surveillance camera was a LEGO element from the LEGO Theme Set (item 9353). The element was placed on top of a vehicle, which moved along gear racks around the outside of the room.

At the end of the day, students wrote down five or six model ideas they wanted to develop as personal projects for the space station.

## Day 4: Personal projects

In the morning of the fourth day, teachers reviewed the model ideas students presented and selected a project each group could finish with the *LEGO TC logo* and computer equipment available. In order to manage the students' requests for extra lights, motors, sensors, and other elements and still provide flexibility for individual projects needs, teachers arranged a grant program for LEGO elements. All groups were given a budget of \$100,000 for extra ele-

ments. Most elements cost \$10,000 apiece. Special elements cost more: lights were \$25,000, motors were \$30,000, and sensors were \$50,000. To acquire extra pieces, students were required to write grant proposals in which they explained how the extra pieces would be used.

While teachers discussed the projects, students completed mural drawings of the space station modules that could be used as backdrops for their space station presentations at the end of the course.

In the afternoon, groups built and programmed their models. Most of these devices incorporated car and conveyor belt activities based on the building instruction cards 9700-3 and 9700-5 in the Technic Control 0 set. Some of the models developed by students include:

- trash compactor
- morse code system<sup>†</sup>
- observatory
- treadmill for exercise
- materials manufacturing system
- food transportation system
- greenhouse
- mail delivery robot
- repair module
- blood purification research laboratory for AIDS patients
- satellite repair maintenance unit

<sup>†</sup> The Morse Code system was inspired by a similar project developed by student Josh Cope of Ladue, MO, shown in the *LEGO TC logo* videotape, which is available on a loan basis from LEGO Dacta. Call 800/527-8339 for information.

## Day 5: Demonstration day

The children affixed their mural drawings on the classroom wall and placed their projects in front. To make demonstrating the models easier, each group's page of programs was copied onto one disk and all the models were connected to two computers in the front of the room. The children presented their projects to parents.

After listening to the student presentations, one of the parents asked if the same course could be run for parents! One student involved in the

class said, "Why can't I stay here and build for 15 days?" A participating teacher said, "I learned so much from these kids that I am ready to build any machine!" ▲

**Manorama (Mano) Talaiver** is a computer/technology educator at the Mathematics and Science Center, 2401 Hartman Street, Richmond, VA 23223, 804/343-6525.

The name LEGO is a registered trademark and the names LEGO Dacta and THE LEGO DACTA CONNEXION are exclusive trademarks of INTERLEGO AG.



*Following is a listing of materials and references Manorama Talaiver used in her Space Station 2000 course.*

**Orientation packages included:**

- 3 yards of aluminum foil
- 12 straws
- 4 red and blue sheets of construction paper
- 12 craft sticks
- 2 folders for cardboard
- 2 ounces of plasticine clay
- 1 color picture of the Space Station from NASA
- 1 booklet entitled *Adventures on Space Station Freedom*, a cartoon-like book produced by TADCORPS, a NASA support contractor for the space station project [Ed. note: unfortunately no longer available].

**Classroom materials included:**

- 6 LEGO TC logo Starter Packs (item 951)
- 5 additional Technic Control 0 building sets (item 9700)
- 2 Technic Teacher Resource Sets (item 9605)
- 3 additional battery boxes using 3 C-size batteries each (2 battery boxes are included in each of the Teacher Resource sets; others can be purchased as spare part items 1335 and 1336.)
- 1 LEGO Elementary School Project Set (item 1053)
- DUPLO® bricks for the verbal directions game (item 9056)
- Videodiscs from Optical Data Corporation, 30 Technology Drive, Warren, NJ 07059. For information on these specific videodiscs, call Optical Data's Customer Service number, 800/524-2481.
- Windows on Science: Life and Physical Science*
- Living Textbook: Life and Physical Science*

**NASA Teacher Resource Centers**

Through the Teacher Resource Centers, educators can receive free information about NASA projects, including lesson plans, videotapes, and NASA project summaries. Below is a partial listing of the Teacher Resource Centers. Contact one near you for more information.

[Ed. note: The following information was correct as of September, 1995.]

NASA Ames Research Center  
Mail Stop TO-25  
Moffet Field, CA 94035  
415/604-3574

NASA Goddard Space Flight Center  
Mail Code 103.3  
Greenbelt, MD 20771  
301/286-8570

NASA Lyndon B. Johnson Space Center  
Mail Code AP-2  
Houston, TX 77058  
713/483-8696

NASA John F. Kennedy Space Center  
Mail Code ERC  
Kennedy Space Center, FL 32899  
407/867-4090

NASA Langley Research Center  
Mail Stop 146  
Hampton, VA 23665-5225  
804/864-3923

NASA Lewis Research Center  
Mail Stop 8-1  
21000 Brookpark Road  
Cleveland, OH 44135  
216/433-2017

Alabama Space and Rocket Center  
Huntsville, AL 35807  
205/544-5812

NASA Stennis Space Center  
Building 1200  
Stennis Space Center, MS 39529  
601/688-3338

## LEGO Lunar Landers

by Michael Lipinski, Millers Falls, MA

I use space exploration as a theme for student problem solving and science experimentation. Two problem-solving tasks I present to my class are "Lunar Landers" and "Operation Egg Drop."

### Engineering challenges

For the "Lunar Landers" activity, I challenge students to build a landing vehicle based on either historic models from the US and the USSR missions or from their own imaginations. Students need not restrict their designs to moon vehicles, but they must design their LEGO landers to meet conditions on the moon or the planet they wish to visit.

Students can create an environment for operating their landers to provide more realism. A few large stones might present interesting challenges to the remote sensing and maneuvering capabilities of their landing vehicles. Can students program the landers to go over or around such obstacles? Can the lander actually scoop up a sample of the planet's soil and store it safely on board? These are the types of problems student engineers must overcome.

### Don't LEGO the egg!

Another popular problem-solving activity is "Operation Egg Drop." Students are given a raw egg to represent an astronaut. The challenge is to package the egg-astronaut so that it survives reentry from space.

Using the LEGO landers students have developed, the class can expand the "Operation Egg Drop" idea and increase the realism of the simulation. Students have already built and tested their LEGO landers. Can they now package their LEGO landers to survive a fall from a building or a greater height? (To prevent the loss of any LEGO elements, it is a good idea to seal the lander in a plastic bag and remove special element sensors, in case the landing is unsuccessful.) If they have been successful, students should be able to cut away the protective packaging, plug their LEGO landers into the computer interface box, and begin exploring.

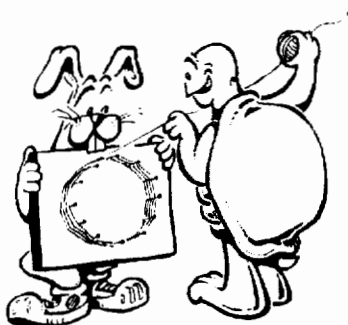
[Ed. note: We pass this information to you from a fellow teacher. This type of space exploration is at your own risk!]

Statement of Ownership, Management, and Circulation (Required by 39 U.S.C. 3685). 1. Publication Title: Logo Exchange. 2. Publication No.: 0888-6970. 3. Filing Date: September 30, 1995. 4. Issue Frequency: Quarterly. 5. No. of Issues Published Annually: 4. 6. Annual Subscription Price: \$34.00. 7. Complete Mailing Address of Known Office of Publication: ISTE, 1787 Agate St., Eugene, Lane, OR 97403-1923. 8. Complete Mailing Address of the Headquarters or General Business Office of the Publisher (Not Printer): ISTE, 1787 Agate St., Eugene, OR 97403-1923. 9. Full Names and Complete Mailing Addresses of the Publisher, Editor, and Managing Editor: Publisher—International Society for Technology in Education, 1787 Agate St., Eugene, OR 97403-1923; Editor—Dorothy M. Fitch, 3 Derby Road, Derry, NH 03038; Managing Editor—David Moursund; 1787 Agate St., Eugene, OR 97403-1923. 10. Owner: International Society for Technology in Education, 1787 Agate St., Eugene, OR 97403-1923. 11. Known Bondholders, Mortgages, and Other Security Holders Owning or Holding 1 Percent or More of Total Amount of Bondholders, Mortgages, and Other Securities: None. 12. The purpose, function, and nonprofit status of this organization and the exempt status for federal income tax purposes: Has Not Changed During Preceding 12 Months. 13. Publication Name: Logo Exchange. 14. Issue Date for Circulation Data Below: Fall 1995 (Volume 14, No. 1). 15. Extent and Nature of Circulation. Average No. Copies Each Issue During Preceding 12 Months. 15a. Total No. Copies (Net Press Run): 1,201 15b. Paid and/or Requested Circulation. (1) Sales Through Dealers and Carriers, Street Vendors and Counter Sales: none. (2) Paid or Requested Mail Subscriptions: 916. 15c. Total Paid and/or Requested Circulation (Sum of 15b(1) and 15b(2)): 916. 15d. Free Distribution by Mail (Samples, Complimentary, and Other Free: 60. 15e. Free Distribution Outside the mail (Carriers or Other Means): 29. 15f. Total Free Distribution (Sum of 15d and 15e), 89. 15g. Total Distribution (Sum of 15c and 15f): 1,005. 15h. Copies Not Distributed. (1) Office Use, Leftovers, Spoiled: 196. (2) Returns from News Agents: none. 15i. Total (Sum of 15g, 15h(1), and 15h(2)): 1201. Percent Paid and/or Requested Circulation: 91%. 16. Extent and Nature of Circulation. Actual No. Copies of Single Issue Published Nearest to Filing Date. 15a. Total No. Copies (Net Press Run): 900. 15b. Paid and/or Requested Circulation. (1) Sales Through Dealers and Carriers, Street Vendors and Counter Sales: none. (2) Paid or Requested Mail Subscriptions: 664. 15c. Total Paid and/or Requested Circulation (Sum of 15b(1) and 15b(2)): 664. 15d. Free Distribution by Mail (Samples, Complimentary, and Other Free: 46. 15e. Free Distribution Outside the mail (Carriers or Other Means): 15. 15f. Total Free Distribution (Sum of 15d and 15e): 61. 15g. Total Distribution (Sum of 15c and 15f): 725. 15h. Copies Not Distributed. (1) Office Use, Leftovers, Spoiled: 175. (2) Returns from News Agents: none. 15i. Total (Sum of 15g, 15h(1), and 15h(2)): 900. Percent Paid and/or Requested Circulation: 92%. 16. This Statement of Ownership will be printed in Volume 14, No. 2 issue. 17. I certify that the statements made by me above are correct and complete. Patti Van Ordstrand, Distribution/Circulation Manager, September 30, 1995.



# The Wonders of String Art

by Jim Muller, CIS Logo Forum



A ball of string can provide a seemingly endless variety of learning experiences for younger children, as described in the article "Learning With a Ball of String" in this issue. It can aid in developing conceptualization and visualization skills as well as logical and structured thought processes. But sometimes putting the string patterns onto the screen can get a bit complicated, especially when you add a few more challenges to the task.

After admiring the box top art that younger children had created, some teenagers decided that they wanted to try their hand at it, but on the screen. The first task was to create a flexible box top procedure.

```
to boxtop :dist :mark
  penup setx :dist - :dist * 2 pendown
  repeat 2 [marker :mark right 90 repeat 2
    [marker :mark] right 90]
end
```

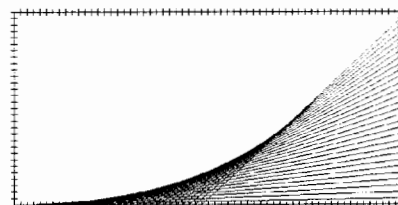
```
to marker :mark
  repeat :mark [forward :dist/:mark marks]
end
```

```
to marks
  left 90
  forward 5 back 10 forward 5
  right 90
end
```

**Boxtop** draws the box top. You define the size of the short side and the number of marks to appear on that side. For example, you might try:

```
boxtop 300 30
```

The next step is creating the string pattern. There is undoubtedly some elaborate algorithm that does the job. The more obvious solution is to move the turtle from one set of coordinates to another. This is what happens with the **curve** procedure.



The **curve** procedure looks complex. But it is simply the turtle doing the sewing that you did with a needle and colored yarn (see "Learning With a Ball of String").

```
curve 30 10 -300 0 300 0
```

You are going to draw 30 lines that are 10 steps apart. You start at :x1 in the lower left corner, where the x-coordinate is -300 and the y-coordinate, :y1, is 0. The turtle moves from :x1, :y1 to :x2, :y2, and then back and forth 30 times.

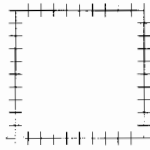
```
to curve :t :gap :x1 :y1 :x2 :y2
  if :t = 0 [stop]
  penup setxy :x1 :y1 pendown
  setxy :x2 :y2
  make "x1 :x1 + :gap
  make "y2 :y2 + :gap
  curve :t - 1 :gap :x1 :y1 :x2 :y2
end
```

Note: If you are using *PC Logo*, add the primitive **list** immediately after **setxy** in both places in the procedure above. If you do not have a **setxy** primitive, use the following tool:

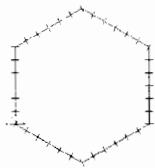
```
to setxy :x :y
  setpos list :x :y
end
```

Box tops are, of course, rectangles. This led to the obvious question about what could be done

with other shaped boxes: hexagons, circles, and the like. Also, what about the ovals and other shaped string and wire art pieces that hang in all the local craft stores?



This led to the search for the “universal pattern procedure.” The one below was as close as we could get. As the surfer continually searches for the ultimate wave, the search for the ultimate pattern procedure continues.

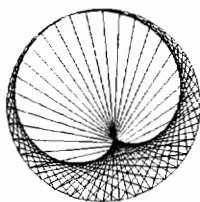


```
to pattern :r :d :m
repeat :r [mark1 :m right 360/:r]
end
```

```
to mark1 :m
repeat :m [forward :d/:m mark]
end
```

```
to mark
left 90
forward :d/20 back :d/10
forward :d/20 right 90
end
```

The next step beyond the pattern procedure was a more exotic circle procedure. This led to a number of different discussions of how and why the **hdg** (heading) procedures did what they did. Was it possible to predict what pattern they would create?



string 150 2 2

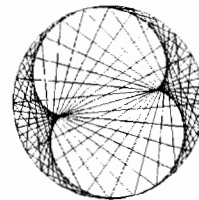
```
to string :rad :d :h
clearscreen hideturtle
make "n 1
```

```
penup setx :rad pendown
repeat 360 [forward :rad * pi/180 left
1]
penup home
repeat 36 * :rad [forward :rad make "p
pos home hdg forward :rad pendown
setpos :p penup home hdg1 make "n :n +
1]
end

to hdg
seth remainder (:n * 5 * :d) 360
end

to hdg1
seth remainder (5 * (:n-1)) 360
end

to pi
op 3.14159
end
```



string 150 3 2

The microworlds within a ball of string have long been a source of fascination. Kindergarten children enjoy creating the patterns as they follow the antics of the turtle. Older children become interested in how planes and objects move through space.

The bottom of the shoebox opened up another world where the flat planes of the box top could be twisted through three-dimensional space. Using the larger shoeboxes opens up the chance to color the string or thread to better demonstrate the twisting, tilting, and moving of planes through space. It is amazing what the imagination can do.

These activities and many others can be found in *The Turtle's Discovery Book*. For information, contact Jim Muller via e-mail at 76703.3005@compuserve.com or on CompuServe at 76703,3005.

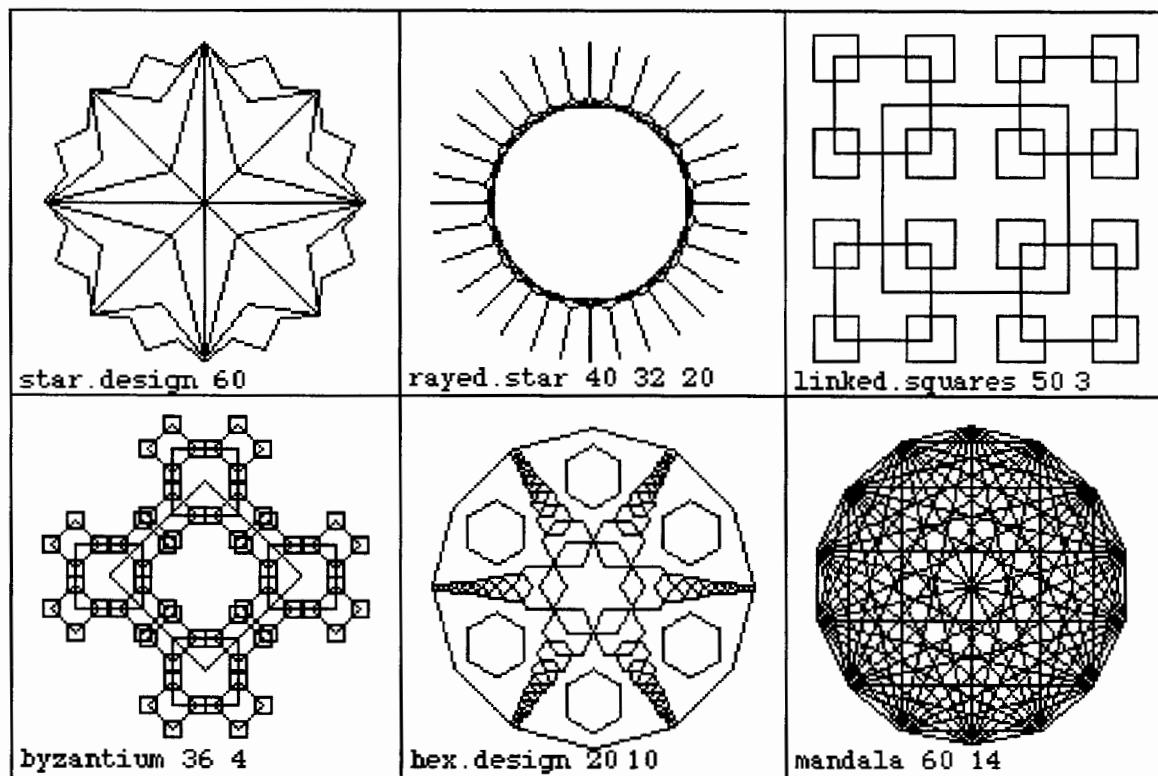


You can read more about **Jim Muller** at the end of his other article in this issue, “Learning With a Ball of String.”



# Turtle On Tour

by Morton Goldberg



At first glance the designs shown above don't look as if they have much in common. However, on careful inspection, you will notice that they are constructed entirely from straight line segments and that regular polygons are major elements.

But inspection, no matter how careful, won't reveal the most important feature they have in common: the Logo procedures that draw them all run a procedure called **poly.tour** at least once. In **poly.tour** the turtle goes on a tour of the vertices of a regular polygon and performs an action at each vertex. What makes **poly.tour** a powerful tool is that the turtle's behavior at a vertex is controlled by an input to **poly.tour**, and thus, may be different each time **poly.tour** runs.

## Thinking about polygons

Actually, I wasn't thinking about geometric designs when I started down the path leading to **poly.tour**. I began by thinking about a different

problem: Given a point, draw an  $n$ -sided regular polygon centered at that point.

With a little knowledge of trigonometry this is an easy problem. The Logo solution is as follows.

[Note: These programs are written using Terrapin's *Logo PLUS for the Macintosh*. They can be run as is with most versions of Logo. If your version does not contain the primitives **distance** or **towardspos**, use the tool procedures listed at the end of this article.]

```
; r = radius of the polygon
; n = number of vertexes—must be an
; integer > 2
```

```
to polygon :r :n
  ( local "turn "angle "side )
  penup
  ; Compute the turtle turn.
  make "turn 360 / :n
```

```

; Compute the vertex angle of the
; polygon.
make "angle 180 - :turn
; Compute the length of a side.
make "side 2 * :r * cos :angle / 2
; Go to the first vertex and turn to
; face the second.
forward :r
right :angle / 2 + :turn
; Draw the polygon.
pendown
repeat :n [forward :side right :turn]
; Undo the initial turn and back down
; to the center.
penup
left :angle / 2 + :turn
back :r
end

```

But I got to thinking: If I pretend I don't know trigonometry, can I still solve the problem? That is, what would **polygon** look like with the restriction that only turtle commands are allowed?

Examine the **polygon** procedure. You will find trigonometry comes into play only when **polygon** computes the length of the polygon's sides in the command

```
make "side 2 * :r * cos :angle / 2
```

How can we eliminate this command? We could replace it with code telling the turtle to perform the following commands:

- Go to the first vertex and record its coordinates.
- Go to the second vertex by way of the center.
- Find the distance from your current position to the recorded coordinates.
- Return to your original position and heading.

Writing this as a Logo procedure gives us:

```

to compute.side
  ( local "x "y )
  ; Go to the first vertex and record
  ; its coordinates.
  forward :r
  make "x xcor
  make "y ycor
  ; Back down to center.
  back :r

```

```

; Turn to face the second vertex.
right :turn
; Go to the second vertex.
forward :r
; Record the distance to (x, y), the
; length of the side
make "side distance :x :y
; Back down from the vertex to
; center.
back :r
; Turn to face the first vertex.
left :turn
end

```

Because we are replacing a command with a procedure, we must also remember to replace

```
make "side 2 * :r * cos :angle / 2
```

in **polygon** with

```
compute.side
```

## Poly.tour

Having disposed of the problem of drawing a polygon without using trig functions, I suddenly realized—in a real Papert mindstorm—that I could draw figures far more intricate than polygons if I instructed the turtle to draw something other than a line segment (i.e., a side of the polygon) each time it visited a vertex.

I started by making a few simple changes to **polygon**. I gave it a new name, **poly.tour**, added an input, **action**, changed the **repeat** command to

```

repeat :n
  [run :action
   forward :side
   right :turn]

```

and removed the **pendown** command that appears before the **repeat**. This gave me the following code for my first version of **poly.tour**.

```

; r = radius of the polygon
; n = number of vertexes—must be an
; integer > 2
; action = run-list, i.e., a list that
; can be given to the run command as
; its input

```



```

to poly.tour :r :n :action
  ( local "turn "angle "side )
  penup
  ; Compute the turtle turn.
  make "turn 360 / :n
  ; Compute the vertex angle of the
  ; polygon.
  make "angle 180 - :turn
  ; Compute the length of a side.
  make "side 2 * :r * cos :angle / 2
  ; Go to the first vertex and turn to
  ; face the second.
  forward :r
  right :angle / 2 + :turn
  ; Send the turtle on a tour of the
  ; polygon's vertexes.
  repeat :n
    [run :action
      forward :side
      right :turn]
  ; Undo the initial turn and back down
  ; to the center.
  penup
  left :angle / 2 + :turn
  back :r
end

```

I say “first version” because it didn’t last long. As soon as I began working with **poly.tour**, I saw that it needed to be improved.

First, to make it easier to use, I made **poly.tour** restore the turtle’s heading after **action** was run and I wrote a tool procedure, **go.to.vertex**, to put the turtle back at the vertex.

Second, **go.to.vertex** needs to refer to the current vertex. Two other tools that I thought would be handy for use within the run-list, **face.outward** and **face.inward**, need to refer to the polygon’s center, so I added the appropriate local names and **make** commands.

What motivated the third and last change is a little subtle. I think it is best illustrated by an example. Suppose you decide to draw a simple design consisting of an *m*-sided polygon with *n*-sided polygons centered at each of its vertexes. You could write the procedure as follows:

```

to mgons.with.ngons :m :n
  poly.tour 100 :m
  [poly.tour 50 :n [ pendown ]
  pendown]
end

```

Although **mgons.with.ngons** looks reasonable, if you were to test it by running

```
right 45 mgons.with.ngons 4 8
```

you would see a square with squares at its corners rather than a square with octagons at its corners. What has gone wrong? The problem is that **poly.tour** has an input named **n**, which has the value 4 when the recursive call to **poly.tour** is made inside the inner run-list. To prevent this kind of accident, you must regard the names of the inputs and locals of **poly.tour** as reserved. Because I didn’t like the idea of simple names such as **r**, **n**, and **turn** being reserved, I added prefixes (**p.** or **v.**) to the names of the inputs and locals of **poly.tour**.

With the improvements incorporated, the Logo code for **poly.tour** looks like this:

```

; p.r = radius of the polygon
; p.n = number of vertexes—must be an
;   integer > 2
; p.action = a run-list, i.e., a list
;   that can be given to the run
;   command as its input

```

```

to poly.tour :p.r :p.n :p.action
  ( local "p.center "p.vertex "p.side )
  ( local "v.angle "v.turn "v.heading )
  penup
  ; Compute the turtle turn—i.e.,
  ; the turn made at each vertex.
  make "v.turn 360 / :p.n
  ; Compute the vertex angle of the
  ; polygon.
  make "v.angle 180 - :v.turn
  ; Compute the length of a side.
  make "p.side
    2 * :p.r * cos :v.angle / 2
  ; Save the turtle's position.
  make "p.center pos
  ; Go to the first vertex and turn to
  ; face the second.
  forward :p.r
  right :v.angle / 2 + :v.turn
  ; Send the turtle on a tour of the
  ; polygon's vertexes.
  repeat :p.n
    [make "p.vertex pos
      make "v.heading heading
      run :p.action
    ]

```

```

    setheading :v.heading
    forward :p.side
    right :v.turn]
; Undo the initial turn and back down
; to the center.
penup
left :v.angle / 2 + :v.turn
back :p.r
end

```

And the code for the three tool procedures looks like this:

```

; Return the turtle to the current
; vertex.

to go.to.vertex
    penup
    setpos :p.vertex
end

; Face the turtle towards the
; center of the polygon.

to face.inward
    setheading towardspos :p.center
end

; Face the turtle directly away from
; the center of the polygon.

to face.outward
    setheading towardspos :p.center
    right 180
end

```

This is a good place to stop discussing **poly.tour** and to start using it. I'll start with two procedures that produce figures (they're too simple to call designs) intended to function as components you can use to build designs.

### Polygon and Stargon

The first figure is a polygon. Now that we have **poly.tour**, the procedure for drawing a polygon becomes trivial.

```

; r = radius of the polygon
; n = number of vertexes—must be an
;     integer > 2

to polygon :r :n
    poly.tour :r :n [ pendown ]
end

```

The second figure is a star polygon. Here **poly.tour** draws an outward pointing angle in place of the side it draws in **polygon**.

```

; r = radius of the star polygon
; n = number of points—must be an
;     integer > 3

to stargon :r :n
    poly.tour :r :n
    [pendown
    right 30
    forward :p.side / sqrt 3
    left 60
    forward :p.side / sqrt 3
    right 30
    penup
    back :p.side]
end

```

### Star Design and Rayed Star

To ease you into the world of turtle tour designs, I'll start with two having particularly short and simple Logo procedures. They also have a common theme: stars.

Originally Star Design was even simpler. It was constructed entirely from stargons and didn't have the eight rays emanating from its center. I added them because there was too much empty space at the center without them.

```

; r = radius of the design

to star.design :r
    ; Draw the quadrant rays.
    poly.tour :r 8
    [face.inward
    pendown
    forward :r
    go.to.vertex]
    ; Draw the nested stargons.
    stargon :r 4
    stargon :r 8
    stargon :r 16
end

```

Rayed Star is constructed from rayed octagons, which is why **n** must be a multiple of eight. It needs at least 24 points to look right—48 points is even better.

```

; r = radius of the star body
; n = number of rays - must be a
;   multiple of 8
; len = length of a ray

to rayed.star :r :n :len
  repeat :n / 8
    [poly.tour :r 8
     [ face.outward
       pendown
       forward :len
       back :len ]
     left 360 / :n]
end

```

## Linked Squares and Byzantium

The theme of next two designs is squares. Both designs appear in James Clayson's book, *Visual Modeling with Logo*. I give this book my highest recommendation. If you like the kind of Logo programming discussed in this article and don't have Clayson's book, put it on your "must-have" list.

The two procedures are nearly the same. They differ only in the **right 45** and **left 45** commands that appear in **linked.squares**. What a difference those two extra turtle commands make!

Also, please note the use of indirect recursion. Both procedures put themselves into the run-list given to **poly.tour**.

```

; r = radius of the central square
; i = down-counter - this counter is
;   reduced by 1 at each stage of the
;   recursion which stops when the
;   counter reaches zero

```

These inputs are the used by both procedures.

```

to linked.squares :r :i
  if :i = 0 [ stop ]
  right 45
  poly.tour :r 4
  [linked.squares :r / 2 :i - 1
   pendown]
  left 45
end

```

```

to byzantium :r :i
  if :i = 0 [ stop ]
  poly.tour :r 4
  [byzantium :r / 2 :i - 1
   pendown]
end

```

Do you wonder where the name Byzantium comes from? The design reminded me of pictures I've seen of jewel-encrusted crosses created in medieval Byzantium.

## Hex Design

What I had in mind when I started working on Hex Design was to draw something that resembled a sand dollar. I arrived at the code you see here after a lot of playful exploration (also known as trial and error or just plain fooling around). I'm happy with the design I've produced, but I call it Hex Design because I don't think it looks enough like a sand dollar to call it one.

```

; r = radius of the central hexagon
; i = down-counter for rays - see
;   hex.ray below

to hex.design :r :i
  ( local "center "distance.to.tip )
  ( local "hex.center "hex.r )
  ; Remember where the center is.
  make "center pos
  ; Draw the central hexagon.
  right 30
  polygon :r 6
  ; Draw the rays.
  repeat 6
    [forward :r
     make "distance.to.tip :r
     hex.ray :r / 2 :i 2
     setpos :center
     right 60]
  ; Draw the hexagons that appear
  ; between the rays.
  right 30
  make "hex.center
    0.7 * :distance.to.tip
  make "hex.r
    0.2 * :distance.to.tip
  repeat 6
    [forward :hex.center
     polygon :hex.r 6
     back :hex.center
     right 60]

```

```

; Draw the outer dodecagon.
polygon :distance.to.tip 12
end

; r = radius of the hexagon to be drawn
; i = down-counter - this counter is
;   reduced by 1 at each stage of the
;   recursion which stops when the
;   counter reaches zero
; k = is the reduction factor applied
;   to the hexagons

to hex.ray :r :i :k
  if :i = 0 [ stop ]
  polygon :r 6
  forward :r
  make "distance.to.tip
    :distance.to.tip + :r
  hex.ray :k * :r / ( :k + 1 )
    :i - 1
    :k + 1
end

```

## Mandala

I've been saving the best for last. Mandala is my favorite design because it's such fun to watch Logo draw it. Try it yourself and you'll see what I mean.

The idea behind Mandala is a simple one. Make a list of the vertexes of a regular polygon and then draw a set of straight lines so that every vertex is connected to every other vertex. In Mandala, **poly.tour**, which has played a starring role in every design up to now, is reduced to a walk-on in which it makes the vertex list. The stars here are **connect.all** and **connect.to.others**, which together perform a double recursion act on the vertex list.

Mandala also differs from the other design drawing procedures in that it always draws its design centered at the origin—it starts (and ends) by forcing the turtle home.

```

; r = radius of the mandala
; n = number of vertexes - must be an
;   integer > 2

```

```

to mandala :r :n
  penup
  home
  connect.all collect :r :n
  penup
  home
end

; Report the list containing the
; vertexes of a regular polygon
;
; r = radius of the polygon
; n = number of vertexes

to collect :r :n
  local "vertexes
  make "vertexes []
  poly.tour :r :n
    [make "vertexes
      fput :p.vertex :vertexes]
  output :vertexes
end

; Given a list of points, draw a line
; between every possible pairing of two
; points in the list.
;
; p.list
;   list of points (coordinates)

to connect.all :p.list
  ( local "p "others )
  make "p first :p.list
  make "others butfirst :p.list
  if empty? :others [ stop ]
  penup
  setpos :p
  connect.to.others :p :others
  connect.all :others
end

; With the turtle at point p, draw
; lines connecting p to all the points
; in p.list.

to connect.to.others :p :p.list
  if empty? :p.list [ stop ]
  pendown
  setpos first :p.list
  penup
  setpos :p
  connect.to.others :p butfirst :p.list
end

```



## Bigger is better

Space limitations forced me to shrink my designs when I made the picture appearing at the beginning of this article. Here are Logo commands that will reproduce them at the correct size:

```
star.design 180
rayed.star 100 48 75
linked.squares 120 4
byzantium 90 5
hex.design 48 16
mandala 180 24
```



**Morton Goldberg** is a freelance programmer and documentation writer. His work requires him to write programs in C++. For recreation and as an antidote, he writes programs in Logo. He can be reached via e-mail at 72346.565@compuserve.com

## Reference

Clayson, J. (1988). *Visual modeling with Logo*. Cambridge, MA: The MIT Press.

### Tool Procedures

If your version of Logo does not include the primitives **distance** or **towardspos**, use the following tool procedures.

```
to distance :x :y
output sqrt ((xcor - :x) * (xcor -
: x)) + ((ycor - :y) * (ycor - :y))
end
```

```
to towardspos :list
make "dx (first :list) - xcor
make "dy (last :list) - ycor
if not :dx = 0 [output 90 - arctan
:dy :dx]
if :dy > 0 [output 0]
if :dy < 0 [output 180]
end
```

If your **arctan** or **atan** primitive takes a single input, remove the line in the **towardspos** procedure above that begins **if not :dx** and replace it with the following lines:

```
if :dx < 0 [output (90 - arctan :dy/
:dx) + 180]
if :dx > 0 [output 90 - arctan :dy/:dx]
```

# Logosium '96

St. Anthony Park Elementary School  
St. Paul, Minnesota  
Friday, June 14, 1996

## Call for Participation

The third annual Logosium will be a day of Logo workshops, discussions, and presentations hosted by the world renowned St. Paul Logo Project. This year Logosium will include sessions conducted by students sharing their projects with other students and with adults.

Sessions may be one-hour presentations or panel discussions, or two-hour, hands-on workshops on any topic of interest to the Logo community.

If you wish to offer a session, submit a one-page description of what you have in mind to:

Marian Rosen & Michael Tempel  
c/o Logo Foundation  
250 West 57th Street, Suite 2228  
New York, NY 10107-2228  
Ph: 212/765-4918  
Fax: 212/765-4789  
e-mail: mbrosen@oui.com  
michaelt@media.mitedu

**The deadline for submissions is  
March 1, 1996**

For registration and hotel information contact:  
NECC '96  
1787 Agate Street  
Eugene, OR 97403-1923  
Ph: 503/346-2834  
Fax: 503/346-5890  
e-mail: necc@oregon.uoregon.edu

Logosium is sponsored by the Logo Foundation and ISTE's SIGLogo.



# High School Geometry Using Logo

by Robert M. Packard

---

In 1986, I was team teaching a class for the University of Alaska entitled “Logo for Elementary Teachers.” My teammate, Charlie Arteaga, emphasized the words and lists portion, and I emphasized the mathematics. After each session Charlie and I would discuss and informally evaluate the effectiveness of the session and finalize plans for the next session.

It was a most enjoyable experience, and as I enthusiastically extolled the virtues of the mathematics embedded in Logo, I began to make statements such as, “A person could use Logo to teach geometry to high school students—it is not just for elementary students.”

Charlie finally issued the ultimate challenge by saying, “Well, why don’t you teach geometry using Logo?”

I proposed the idea to my principal and the local board of education. I had recently been awarded the Presidential Award for Excellence in Teaching Mathematics and my credibility was such that the course was approved.

The approach that first year was casual and loose. Students seemed to enjoy the class and were eager to accept the mathematical challenges I offered. The course seemed more like a new toy than a serious mathematics course. I moved to Colorado and the course came with me.

Slowly the course began to evolve on its own. Some of the tools from a traditional analytic geometry course became quite useful. I wanted the course to have a lot of open-ended questions—to be a guided discovery course in geometry. I felt free to try almost anything. Exercises, lessons, a variety of activities were tried, some of which were retained and some discarded. The course needed some structure, yet remain free enough to encourage a wide range of exploration in geometry.

When a student discovers a substantial theorem on his or her own, that theorem carries the

student’s name for the year. We refer to that theorem as the “McGarry Theorem” or whatever the student’s name happens to be. It doesn’t happen often, but it does become a motivating factor. Each student is made to feel that a theorem with his or her own name is within reach.

## Beginning activities

The course begins with an exploration of line segments and angles. Exercises include simple ones, such as writing a name or drawing a house with a chimney and trees. We use just a few primitive commands: **forward**, **back**, **right**, **left**, **penup**, **penerase**, etc. The angle as a measure of rotation is difficult for many high school students, so they need time to explore this idea.

We continue to explore concepts by introducing the Cartesian plane and the **setxy** command. We move around the screen using ordered pairs and draw polygons using the **repeat** command as well as **setxy**. We draw and name many types of polygons, both regular and common. In drawing and naming these polygons, students begin to discover some of the basic characteristics of polygons. We bring the focus to triangles of all types and begin to zero in on right triangles. I believe students learn much from trying to draw a right triangle using **forward** and **right** commands.

## Pythagorean theorem

The Pythagorean Theorem is the next natural step. We develop it informally by drawing squares on each of the two legs of the right triangle as well as on the hypotenuse. We draw a regular pentagon on each of the sides of the right triangle, and we use the **fill** command and look at the areas. We spend time drawing regular hexagons, octagons, parallelograms, and all sorts of similar figures with sides that correspond to the sides of the right triangle. As students become visual believers in the Pythagorean Theo-



rem, we spend time relating the concept to the symbols of  $a^2 + b^2 = c^2$ .

Because the Pythagorean Theorem is a cornerstone of our geometry course, we take much time and effort exploring it.

### Analytical geometry

After the Pythagorean Theorem, our course takes on a decidedly analytical geometry flavor. The distance formula is developed along with the midpoint formula and a technique for finding the fractional distance from any one point to another. We look at medians, altitudes, perpendicular bisector of segments, and angle bisectors, particularly as they relate to triangles. That particular section ends with an exploration of Euler's Line.

Slope is introduced from the basic definition, even though the students were exposed to slope in Algebra I, a prerequisite for geometry. Analytic geometry provides an excellent vehicle for relating slope, parallel lines, and perpendicular lines. At this point we introduce some traditional deductive proofs and continue those through congruent triangles.

Similar triangles and polygons along with ratio and proportion are explored with a culminating activity of drawing house plans to scale, complete with furniture and outside landscaping.

Circles and the theorems relating to circles are investigated by using a procedure that draws a multisided polygon. Discussions regarding representations of lines and segments are held again. I wanted a circle to be the locus of points in a plane that are equidistant from a fixed point called the center, hence the procedure. The procedure, one of a very few that we use is as follows:

```
to circle :radius
hideturtle
repeat 20 [penup forward :radius right
  90 pendown forward :radius/6 back
  :radius/3 forward :radius/6 left 90
  penup back :radius right 360/20
  pendown]
showturtle
end
```

### Other activities

A few capable students learn to solve quadratic equations with real roots using only a compass and straightedge. This activity helps them make connections with irrational numbers.

The section on transformational geometry includes reflections in lines and points, dilations in both lines and points, and rotations about a point.

Another unit explores tessellations. The students write procedures to tessellate a polygon over the screen. They examine which polygons will tessellate before we begin slides and rotations. Some students are capable of exploring the Hilbert curve.

Students have the option of investigating right triangle trigonometry using sine, cosine, and tangent and arctangent through exercises that are traditional, yet exploratory.

Students can also explore fractals, recursion, and chaos. We play a chaos game on the computer. We also build some Sierpinski triangles using a repeating pattern with graph paper and pencil.

Students seem to enjoy the geometry course very much. Logo is an excellent tool for helping deliver mathematics in a way that allows students to explore mathematics in an environment that is not pressured or threatening. We at Florence High School are having very good success with the mixing of Logo and high school geometry. ▲

**Bob Packard** teaches secondary math at Florence High School, Florence, Colorado. He can be reached via e-mail at [BobPackard@aol.com](mailto:BobPackard@aol.com).

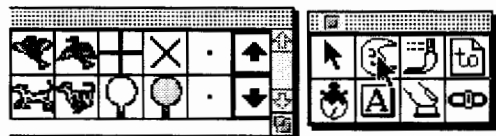
# Control Simulations (Part 2): Fan and Thermostat

Glen L. Bull, Gina L. Bull, and Todd Kent

In the previous issue, we constructed a simulated lamp and temperature sensor in *MicroWorlds*. In this column, we will add two more devices: a fan and a thermostatic control.

## Lamp and temperature sensor: synopsis from Part 1

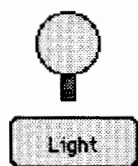
The lamp created in the previous column was based on two shapes: one empty and one filled with yellow (to simulate a lighted lamp).



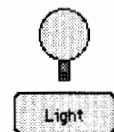
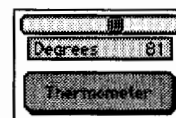
A procedure was developed on the procedure page to toggle a turtle (assigned the name "Light") between the two shapes.

```
To Light
  Talkto "Light
  If Shape = "Off [Setsh "On Stop]
  If Shape = "On [Setsh "Off Stop]
End
```

A button was created that could turn the lamp on and off by clicking it. (The instruction assigned to the button was the name of the procedure above; this instruction also appears as the button label.)



A slider bar was then used as the display for a thermometer that monitored the simulated temperature when the lamp was turned on.



The Thermometer procedure assigned to the Thermometer button and two associated procedures were developed in the following manner:

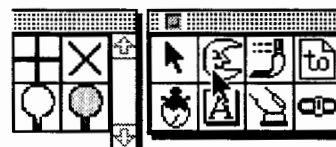
```
To Thermometer
  If Light? [IncreaseDegrees]
End

To IncreaseDegrees
  If Degrees > 98 [Stop]
  Set "Degrees "Value Degrees + 1
End

To Light?
  Light, Output Shape = "On
End
```

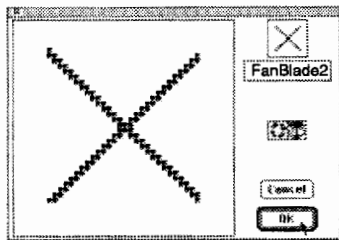
## Constructing a simulated fan

The simulated lamp constructed in Part 1 heated the temperature sensor. The next step is to create a simulated fan to create a breeze to cool the temperature sensor. Two shapes—a cross and an "X" shape—will simulate different phases of rotating fan blades. Click the moon icon in the tools palette to access the shapes center.



We assigned the name "FanBlade1" to the cross shape and "FanBlade2" to the "X" shape we created. The arms of the vertical cross

(FanBlade1) should extend the full height and width of the shape editor, touching the sides of the box enclosing it. However, the arms of the second shape (the "X" shape named FanBlade2) should not extend all the way to the corners of the box. (Because the tips of rotating fan blades travel in a circle, they will touch the tops and bottoms of a box in which they are enclosed, but not the corners. The illusion of a rotating fan blade will be enhanced if this is taken into account when the two fan blade shapes are constructed.)



Once the two fan blade shapes have been constructed in the shapes editor, it will be necessary to hatch a second turtle to serve as the fan. Select the "Turtle Hatching" icon in the tools palette.



Then click anywhere on the page to hatch a second turtle.



Once the turtle appears, click on the pointer tool in the tools palette to select it.

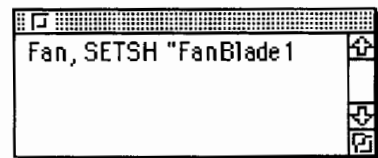


Hold down the Shift key, and use the pointer tool to double-click the newly hatched turtle to produce its dialog box.



When the turtle's dialog box appears, enter the name "Fan" in the name box. Then click OK to exit the dialog box.

Enter the command shown in the illustration below to change the shape of the turtle into the first fan blade shape. By alternating between the two fan blade shapes, the illusion of motion can be created.



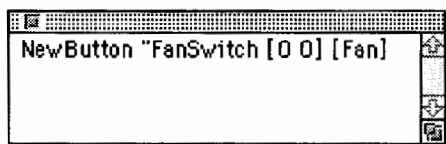
We used the paint tools to create a fan housing consisting of a circle around the fan blade and an external square, and used the paint bucket to fill the space between with gray. You can determine how much or how little detail of this kind you incorporate into your simulation.



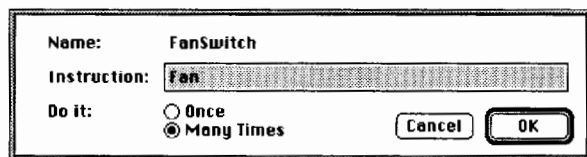
After creating the fan, add the following procedure to the Procedures page. This procedure shifts the fan blades between the two alternate shapes. We found that we needed to add a short wait. Depending on the speed of the your computer, you may or may not need to slow down the movement of the alternating fan blades in this fashion.

```
To Fan
  Talkto "Fan
  Setsh "FanBlade1
  Wait 1
  Setsh "FanBlade2
  Wait 1
End
```

After entering the Fan procedure, return to page1. Create a new button named "FanSwitch" by entering the following command in the command center, and pressing Return.



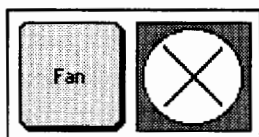
After creating the button by entering the above command, hold down the Shift key and double-click the new button to produce its dialog box. Verify that the name of the button is FanSwitch and that its instruction is Fan. If that is the case, click the Many Times option to select it, and then click the OK button to exit the dialog box.



If you want to change the dimensions of the Fan button, hold down the Shift key and click the button once. This will produce a handle in each corner. Drag a handle in the desired direction to resize the button. Click outside the button to deselect it; the handles will disappear.



Drag the Fan button beside the fan. Clicking the Fan button once should turn the fan on; clicking it a second time should turn it off.



The turning fan blades will not affect the simulated thermometer until the Thermometer procedure is revised. Add the procedures DecreaseDegrees and Fan? to the Procedures page, then edit the Thermometer procedure by adding the line, If Fan? [DecreaseDegrees], as in the following example:

```
To Thermometer
  If Light? [IncreaseDegrees]
  If Fan? [DecreaseDegrees]
End
```

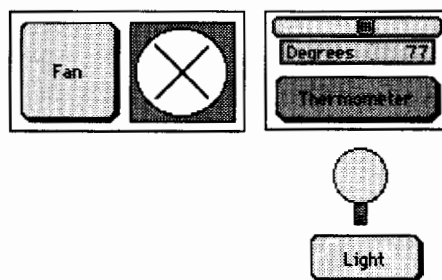
```
To DecreaseDegrees
  If Degrees - 2 < 33 [Stop]
  Set "Degrees "Value Degrees - 2
End
```

```
To Fan?
  Output Get "FanSwitch "On?
End
```

If the light is on, the Thermometer procedure increases the simulated temperature. If the fan is on, it decreases the temperature. The Fan? procedure checks the state of the button named FanSwitch. If the state is On, the Thermometer procedure decreases the value of the Degrees slider.

The DecreaseDegrees procedure drops the temperature twice as fast as the lamp increases it when both the light and fan are on. In other words, the breeze from the fan is assumed to override the heat from the lamp. Other values could be assigned to the heat from the lamp and the cooling effect of the fan, changing the assumptions underlying the simulation.

After adding the two additional procedures to the Procedures page and revising the Thermometer procedure, return to page1. Be sure the Thermometer procedure is running. (The Thermometer button will be highlighted if it is; if the Thermometer button is not highlighted, click the button once to start it again.) Then click the Fan button to turn the fan on.



The fan should cool the thermometer, dropping the temperature displayed.

## Constructing a thermostatic control

Once the light, the fan, and the thermometer have been constructed, creating a thermostatic procedure is straightforward. In a home, a thermostat provides a control that sets the point at



which the heating or air conditioning will come on. The Thermostat procedure in Logo checks to see whether the Degrees value, measured by the thermometer, exceeds a set point. When that is the case, the fan is turned on. When the thermometer is cooled and drops below the set point, the fan is turned off. Add the following procedures to the Procedures page, then return to page1.

```
To Thermostat
  If Degrees > SetPoint [FanOn]
  If Degrees < SetPoint [FanOff]
End
```

```
To FanOn
  Set "FanSwitch "On? "True
End
```

```
To FanOff
  Set "FanSwitch "On? "False
End
```

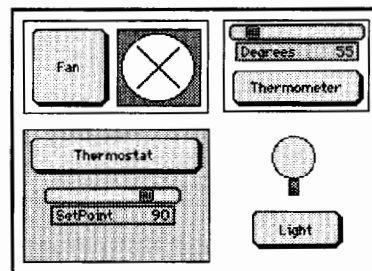
Then create a button named "Thermostat" and a slider bar named "SetPoint." To accomplish this, first select the slider tool in the tools palette.



After selecting the slider tool, click anywhere in the page. A dialog box should appear. Enter the name "SetPoint" in the name box and enter minimum and maximum temperatures. (We are using the values of 32 degrees and 99 degrees.) After entering this information, click the OK button. The slide bar created in this way will provide a means of setting the temperature on the simulated thermostat.

Then select the button tool from the tools palette and click anywhere on the page to create a button. A button dialog box should appear. Enter the procedure name "Thermostat" in the instruction box and select the Many Times option. (It is crucial to select Many Times or the procedure will not operate properly.) Once this is done, click the OK button to exit the dialog box.

Set the value of the SetPoint slider bar to the point at which you would like the fan to come on as the thermometer heats up. Begin with the fan off and the light on. The thermometer and thermostat procedures should be activated (i.e., these buttons should be highlighted). The degrees registered on the simulated thermometer will increase as the light heats it up. When the temperature reaches the set point established on the thermostatic control, the fan should come on.



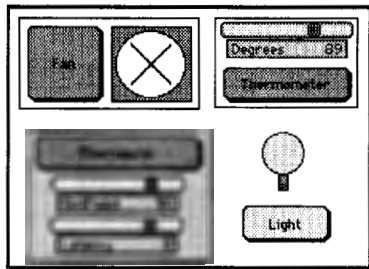
In the illustration above, the temperature of the thermometer (shown as 55 degrees) will increase until it reaches the set point of the thermostat (90 degrees). At that point the fan will turn on, cooling the temperature until it drops below the set point.

In this example, the fan will repeatedly cycle on and off, cooling the temperature a degree or two below the set point, cycling off, and then cycling back on again as soon as the temperature rises one or two degrees. This "hunting" phenomenon is sometimes observed in the thermostat of a home cooling system if the sensitivity is not set properly. If the temperature is confined to too small a range, the system will repeatedly cycle on and off throughout the night, keeping everyone awake.

One possible solution is to adjust the thermostat so the temperature drops several degrees below the set point before the system turns on again. If the range is too wide, the inhabitants of the house may become uncomfortable before the system responds. If the range is too narrow, the system will turn on and off too often.

In the case of the simulation, a slider bar that controls the latency of the response (i.e., how long the fan waits before turning on again) can be added to the thermostatic control. Normally this control is adjusted by a service technician

rather than by a home owner, but in our example we've placed it on the front control panel to make experimentation easier.



Only a small change to the Thermostat procedure is needed to incorporate the latency control. The latency can be subtracted from the set point in the Thermostat procedure, as shown below:

```
To Thermostat
  If Degrees > SetPoint [FanOn]
  If Degrees < SetPoint - Latency
    [FanOff]
End
```

## Comparisons and bells & whistles

The LEGO corporation has developed a product with motors and external temperature sensors that can be connected to the computer to create a working model of a thermostatic control. One of the first examples described in the *LEGO Control Lab* consists of a fan with a thermostatic control. Since the *Control Lab* software and *MicroWorlds* were both developed by Logo Computer Systems, Inc., it is not surprising that the Logo language underlies both.

If a *LEGO Control Lab* kit is available, comparing the simulated thermostat with the working model would be a useful educational exercise. However, the simulated thermostatic control developed in *MicroWorlds* can open the door to a variety of worthwhile inquiries in its own right.

We included several other bells and whistles in our thermostatic simulation. For example, we added a speed control to the fan (high, medium, and low speeds) to allow experimentation with the simulated cooling effect of different speeds. The types of refinements that your students might add would depend upon the direction that their research takes them, but other enhance-

ments typically suggest themselves as projects of this kind are implemented.

## Summary

Since the introduction of *HyperCard* in 1987, a number of programs have been developed that support development of hypermedia activities that take the form of clickable objects linked to various forms of media such as sounds, images, and animation. These multimedia capabilities offer great educational potential. However, *MicroWorlds* offers more than multimedia capabilities, as valuable as they are. Often the potential of *MicroWorlds* that goes beyond multimedia lies untapped.

We populated one "microworld" with several simulated devices as an illustration of some of the possibilities. Other simulations might involve control systems of household appliances, or perhaps a biologic lifecycle. A casual observer might not notice much difference between this simulation and a point-and-click multimedia program. The value of the simulation lies in its construction by students, who acquire and experiment with sophisticated concepts through the development process. The limitations of your microworlds are limited only by the curiosity of your students. ▲

**Glen Bull** is an associate professor in the Instructional Technology program of the Curry School of Education at the University of Virginia. **Gina Bull** is a computer systems engineer in the Information Technology and Communication organization at the University of Virginia. **Todd Kent** is a graduate instructor in the Curry School specializing in instructional uses of technology.

GBull@Virginia.edu  
Gina@Virginia.edu  
TKent@Virginia.edu



# Logo: Search and Research

## Does Logo Learning Increase Over Time?

by Douglas H. Clements and Julie Sarama

---

In the last column, we saw that work with Logo can have positive long-term effects. Some studies suggest an even more powerful and surprising finding: *greater* effect in the long term than the short term! On the basis of traditional learning theories, this should be impossible. So, what's happening?

### Increasing effects over four years

Janice Flake and Marvel Lou Sandon (1990) became interested in tracking several groups of students over the years. Each of the 48 students was with one of the school's four fourth- and fifth-grade teachers for both of those years. All of the students had the same sixth- and seventh-grade mathematics teacher. So, they received the same instruction for grades 6 and 7.

In their fourth- and fifth-grade years, all four classes worked with measurement. The *single difference* was this: Only one of the four fourth- and fifth-grade classes worked with Logo.

To trace their achievement, the researchers examined the students' standardized test scores and state assessment for grades 3 through 8. The students could not be distinguished during their fourth and fifth grades ... the two years that one class worked with Logo.

However, scores from their middle school years showed that the Logo students outperformed the non-Logo students in mathematics. Why? The researchers believe that their work in earlier grades contributed to a major foundation in mathematical imagery. With such imagery, students could grasp the concepts of fractions, decimals, and percents in the middle school grades. Note that the researchers do not have specific data supporting this belief that imagery was the actual mechanism. The scores, however, are evidence that Logo had a positive delayed effect.

### Not higher on immediate posttest, but significantly higher on delayed posttest

We are also interested in the long-term effects of Logo work. In one study, we investigated fifth graders' learning of symmetry and geometric motions: slides, flips, and turns (Johnson-Gentile, Clements, & Battista, 1994). We had three groups. Two were treatment groups. One used our Logo microworlds (Clements & Meredith, 1994, is our latest version.) The other used manipulatives and paper and pencil. Both of these treatment groups worked on the same eight lessons on geometric motions and symmetry. The only difference was that one used Logo, the other used manipulatives. The third (control) group used a textbook's lessons.

Interviews revealed that both of the treatment groups performed at a higher level of geometric thinking than the control group (we're referring to the van Hiele levels that we've discussed in previous columns). Both treatment groups outperformed the control group on an immediate posttest. They took this test immediately after their instruction on geometric motions. Both treatment groups also outperformed the control group on the delayed posttest, which they took a month later—with no work on motions and symmetry in that month.

OK, so the lessons were good. And working with materials—Logo or manipulatives—was better than working with a textbook. Nice to hear, but no huge surprise. The more interesting question is this: How did the two *treatment* groups compare? On the immediate posttest, they did not significantly differ from each other, though the Logo group scored a bit higher than the manipulatives group. However, the Logo group significantly outperformed the manipulatives group on the *delayed* posttest. So, the Logo-

based version of the lessons led to better long-term learning.

Again, why? We believe that interaction with the Logo microworlds enhanced the students' construction of higher-level conceptualizations of motion geometry. What is our specific evidence?

The Logo group's scores significantly *increased* from immediate to delayed posttest. Those of the manipulative group significantly *decreased*. This resulted in the significant difference in favor of the Logo group on the delayed posttest.

How can scores increase without new experiences? Investigation of individual items suggested specific reasons. On certain tasks, Logo group scores increased, and manipulatives group scores decreased. Several of these tasks involved figures that had more than one line of symmetry or parallelograms with no lines of symmetry. Logo students appeared more likely to identify horizontal and oblique lines of symmetry and less likely to be misled by the rotational symmetry of the parallelogram. What does this mean? It suggests that Logo students could better resist intuitive visual thinking and apply analytical thinking instead.

In a similar vein, scores of the Logo group increased on items requiring the identification of motions and their properties (i.e., turn centers). So, differences on the delayed test can be attributed to Logo students' construction of higher-level concepts. That is, Logo students analyzed components and properties of geometric motions.

Other evidence supports this notion. For example, one interview item asked students to tell what it means to say that two figures are congruent and how they would know for sure that they are congruent. The responses of the control students were at a low level.

- Five said they would measure but did not describe what would be measured.
- Two mentioned "putting one on top of the other."
- Five said they didn't know or gave inconsistent or uninterpretable responses. For example: "If one was smaller and one was bigger, they would be congruent."

All of these were level 0 responses for the domain of motion geometry. Recall that level 0 means little or no understanding of the idea. Responses of the manipulatives students were more sophisticated. However, they were closely tied to a procedural description involving the manipulatives used during the treatment.

- Seven manipulatives students said that they would prove congruence "by taking a piece of tracing paper, tracing one shape, and then putting it on the other" (a level 1 response—recognition of the idea, but little knowledge of the properties or specifics of the data).
- Five students explicitly mentioned motions and their relationship to congruence (a level 2 response—knowledge of properties and relationships). For example: "...by flips, slides, and turns you could move one figure onto another to see if they fit. It's important to check by putting one piece onto the other."

Logo students' descriptions were at a higher level and were more general and abstract.

- One said he would measure the sides and angles to see if they were all the same (level 2).
- Three said they would put one figure on top of the other to see if they fit (level 1 response, but not tied to Logo).
- Eight Logo students explicitly mentioned motions and the motions-congruence relationship, but did not mention computers or specific computer procedures (level 2 and abstract). For example: "You could use a series of motions—flips, slides, and turns—to try to move one figure onto the other. If they match exactly, they're congruent."

On four of the six items, the Logo group responded at higher levels of thinking than did the manipulatives group (making more level 2 responses).

Why? The need in Logo environments for more complete, precise, and abstract explication may account for students' creation of conceptually richer concepts for motions. That is, in Logo, students have to specify steps to a noninterpretive agent, with thorough specification and detail. The results of these commands can be observed, reflected on, and corrected.

In manipulative environments, one can make intuitive movements and corrections without explicit awareness of geometric motions. For example, even young children can move puzzle pieces into place without conscious awareness of the geometric motions that describe these physical movements. In the noncomputer environment used here, attempts were made to promote such awareness, but descriptions of the motions were generated from and interpreted by the physical motions of students. The students also understood the task and thus interpreted the descriptions in that context.

Observations during the lessons indicated that Logo students were more likely to discuss the motions themselves, treating the motions as concrete processes and objects of reflection, as opposed to discussing the figures or manipulatives undergoing the motion.

Recall that the only difference between the treatments was the use of the Logo environments. This difference existed for only six of the eight lessons. Logo can make a significant contribution to higher-level thought—especially over time! It can serve as a conceptual framework for developing sophisticated ideas. ▲

**Douglas H. Clements**, Professor at the State University of New York at Buffalo, has studied the use of Logo environments in developing children's creative, mathematics, metacognitive, problem solving, and social abilities. Through a National Science Foundation (NSF) grant, he developed a K–6 elementary geometry curriculum, *Logo Geometry* (published by Silver Burdett, & Ginn, 1991). He is currently working with several colleagues on a second NSF-funded project, *Investigations in Number, Data, and Space*, to develop a full K–6 mathematics curriculum featuring Logo. With Sarama, he is co-authoring new versions of Logo for learning elementary mathematics. One, *Turtle Math*, is available from LCSi.

**Julie Sarama** is a mathematics education doctoral student at the State University of New York at Buffalo. She has taught secondary mathematics and computer science, gifted math at the middle school level, and mathematics methods courses. Along with Clements, she is co-author of *Turtle Math* and is currently designing and programming new versions of Logo.

Douglas H. Clements and Julie Sarama  
State University of New York at Buffalo  
Department of Learning and Instruction  
593 Baldy Hall, Buffalo, NY 14260  
E-mail: Clements@ubvms.cc.buffalo.edu

Time to prepare this material was partially provided by the National Science Foundation grant numbers ESI-9050210 and MDR-8954664. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation.

## References

- Clements, D. H. & Meredith, J. S. (1994). *Turtle math* (Computer program). Montreal, Quebec: Logo Computer Systems, Inc. (LCSI)
- Flake, J. L. & Sandon, M. L. (1990). Longitudinal study of mathematics achievement for a technological and experiential based mathematics curriculum. *Journal of Computing in Childhood Education*, 2(1), 55–67.
- Johnson-Gentile, K., Clements, D. H., & Battista, M. T. (1994). The effects of computer and noncomputer environments on students' conceptualizations of geometric motion. *Journal of Educational Computing Research*, 11, 121–140.

# Global Logo Happenings

Edited by Dennis Harper

## Logo Exchange Continental Editors

### Africa

Fatimata Seye Sylla  
UNESCO/BREDA  
BP 3311 Dakar  
Senegal, West Africa

### Asia

Marie Tada  
St. Mary's Int. Sch.  
6-19 Seta 1-Chome  
Setagaya-Ku  
Tokyo 158, Japan

### Australia

Anne McDougall  
Monash Univ.  
6 Riverside Dr.  
East Kew  
Victoria 3120  
Australia

### Europe

Harry Pinxteren  
Logo Cent. Nederland  
P.O. Box 1408  
BK Nijmegen 6501  
Netherlands

### Latin America

José Valente  
NIED  
UNICAMP  
13082 Campinas  
Sao Paulo, Brazil

This edition's International column is written by José Valente of Brazil. Few educators in the world have used Logo more humanely than Dr. Valente. This article about the use of Logo with the street kids of Brazil is only a summary of a longer article describing in detail the work of Dr. Valente and his colleagues. Please e-mail Dr. Valente or me ([dharp@chs.osd.wednet.edu](mailto:dharp@chs.osd.wednet.edu)) for the complete article or further information.

## The Use of Logo With Disadvantaged Children

*by José Armando Valente*

### Introduction

The use of Logo in special education has promoted the development of an education that is truly special. The Logo methodology has made possible the creation of computer-based learning environments for different populations of special needs children (Valente, 1992). These children were able to learn and develop intellectually, thanks to an educational methodology adapted to their specific need, in a learning environment enriched by the presence of the computer.

Just as the computer has fostered the creation of learning environments that produce impressive results in the intellectual development of handicapped children, it has been used to enrich the learning environment of disadvantaged children—or “street kids,” as they are more commonly known.

This article describes the experience of the project “Educação Científica para os Meninos de Rua de Brasília” (Scientific Education for Street Children of Brasília), which introduced the computer in these children's educational setting. Project PROEM, Promoção Educativa do Menor

(Educational Promotion of the Minor), is described and proposed as an alternative educational setting for this population, followed by an explanation of how Logo is used as an educational tool at PROEM. Finally, the article ends with a discussion and concluding remarks.

### The PROEM

The project PROEM is connected to the public educational system of the Federal District of Brasília. The project was established during the 1980s with the objective of providing assistance to the disadvantaged children who were living in the streets of the city of Brasília. All the children enrolled in PROEM are from poor families living in towns around Brasília.

PROEM has as its educational model the Open School, and presents a unique pedagogical proposal—an individualized teaching method with specific attention to the needs of each child. The educational content is adapted to the reality of the PROEM population and the program is geared to prepare these children vocationally. The pedagogical proposal is based on the content proposed by Piaget's constructivism approach, which is coherent with the learning process that these children use to survive in the street—knowledge is not taught directly but is the product of the child's own experience in the street.



A project using computers in the school activities was initiated in order to awaken and maintain the students' interest in learning and provide them with a background in which they could acquire vocational skills. Thus, the project "Educação Científica para os Meninos de Rua de Brasília" was established in the school in 1989.

The pedagogical basis for the computer project is in line with the constructive approach adopted by PROEM. This means that the computer is used as a tool instead of as a teaching machine. Thus, the Logo methodology was chosen along with a few pieces of software, such as a word processor and spreadsheet to be used by the students.

### **The Logo learning environment in action**

PROEM, as an alternative school, views the curriculum as a guide and not the principal object of the teaching process. To a certain extent, this is imposed by the structure of the school: the grades are grouped, the student's intellectual level is too heterogeneous, and each student's needs require an individualized type of work. Accordingly, the teacher prefers to develop themes that allow for the exploration of a particular content and the integration of several disciplines, instead of following the curriculum strictly. Logo was integrated into this learning process.

Logo is used to develop activities that are more specific, such as the elaboration of texts in Portuguese, development of specific projects in mathematics such as the study of area measurement, and projects in science such as modeling of the respiratory system and the movement of objects according to Newton's laws of physics.

During the academic year all of the groups used Logo in some aspect of the curriculum. This work was conducted by the teacher of a particular discipline. The students used Logo two to three hours per week, depending upon each group's level of interest and knowledge about computers. Based upon observations, interviews, and analysis of the student behavior, a member of the school's administration staff stated that she has noticed an elevation in the level of self-esteem in the majority of the students and in

the students' interest in entering the working market place. She hypothesized that this was attributable to the students' use of Logo. According to a study done in the school, about "90% of the students who received a general vocation initiation got a formal job in the computer area, increasing, thus, the demand for courses of this nature" (Macedo & Suguri, 1992).

### **Discussion and conclusion**

The existence of boys and girls living in the streets is a new phenomenon in our society. Some of these children, who are victims of aggression in their infancy, rebel against this society and become aggressors. Victims of aggression and aggressors are perplexed, shocked, and frightened; we are living in a society of fear. With all the technological, economical, social, and intellectual progress, society proves its incompetence by being incapable of solving the disadvantaged children's problem.

This tragic and devastating situation is not unique to Brazil. According to data from the United Nations Fund for Infancy, in the world "approximately 30 million children live in the streets, children who run away from their house, who are abandoned, or who are orphans. The majority are deprived of health care and of education and almost all are led to face the hard choice of resisting or joining the violence, crime, prostitution and the use of drugs, which are part of the reality of street life in cities like Lima, Rio de Janeiro, Bombay, Lagos and New York" (UNICEF, 1991, p. 36).

The problem is very complex and concrete. Any solution proposed has to consider education. However, the education has to be such that it provides a substantial qualitative leap sufficient to overcome the stigmas and the schooling delays the children may have experienced during their street life. Such a leap is possible through the usage of frontier technology, such as computers. Thus, computers and Logo can have an important role in the disadvantaged child's learning process.

But, Logo alone is not a solution. Effective solutions are related to the understanding of the real problems of the disadvantaged child. First, the disadvantaged child is identified or

commonly referred to as "street kid," who has no family or home, and who exists by stealing and by dealing in drugs. This stigma is debilitating to the child on whom it is imposed as well as to the society that labels the child. The child is marginalized, and society refuses to release the child from this stigma, even in cases of absolute success.

Second, the emphasis of the studies of this population, up until now, has been on the socioeconomic identification of the disadvantaged child. Very little is known about their intellectual and emotional development. This lack of knowledge, together with the assumption that all disadvantaged children are "street kids," has led us to the adoption of ineffective educational solutions.

The idea that disadvantaged children have learning deficiencies needs to be revised. Evaluating the disadvantaged child requires a more creative approach than evaluating children from traditional home settings. Disadvantaged children have acquired, through their experience in their environment, a level of knowledge that is very sophisticated in some areas, such as the ability to survive in the street. The traditional evaluation tests only emphasize the academic aspects of performance. Therefore, it is necessary to evaluate these children's intellectual potential with activities they want to develop, activities that are related to the reality in which they live.

Finally, the actual educational system is very unprepared to deal with the disadvantaged child population. The lack of knowledge about this population does not provide the teachers with the means to penetrate into the world of these children. The teachers, thus, have great difficulty in getting to know the children, establishing an intellectual relationship with them, and creating a learning environment to meet their needs. Also, the teachers do not have the proper pedagogical resources to facilitate the learning about the children's cognitive and emotional development. In this sense, the work with the computer has created conditions that alert teachers to the necessity of getting to know the learner so as to assist in his or her learning process.

However, the work with Logo that has been proposed does not mean simply the addition of

the computer in the classroom. Logo had a very important role in creating a learning environment at PROEM. Obviously, the computer facilitates and professionalizes the drawing process. However, the students' interest in using Logo to produce drawings allowed them to get in touch with spatial concepts and problem-solving techniques. Logo activities allowed the teachers to understand the different ways these drawings were being produced and, therefore, to get to know each one of the students. Thus, Logo allowed the students to obtain a product that has, first, an "intellectual signature," represented by the student's knowledge, style, and creativity; and second, involved the student in an activity that was challenging, motivating, and valued by society.

Of course, this project can be replicated in other institutions in Brazil and in institutions in other countries. The only requirement is that the institution as well as the teachers must be interested in promoting educational changes. The institution must want to implement an education that is alternative and takes into consideration the students' interests and needs. The teachers also must alter their traditional attitude and, instead of teaching, promote the creation of learning conditions. Thus, these teachers must be trained pedagogically and technologically to be able to use Logo as a tool for the students to develop academic tasks. ▲

#### **José Armando Valente**

Coordenador of Núcleo de Informática Aplicada à Educação – NIED

Universidade Estadual de Campinas – UNICAMP  
Cidade Universitária, Prédio V de Reitoria – 2º Piso  
13083-970 Campinas, SP – Brazil

Fax: ++55 (192) 39 4717

E-mail: jvalente@ccvax.unicamp.br

#### **References**

- Macedo, A. L. M. & Suguri, V. L. A. (1992). *Relatório de informática do PROEM*. Unpublished manuscript, PROEM, Brasília.
- UNICEF. (1991). *Situação mundial da infância 1991*. Brasília: Fundo das Nações Unidas para a Infância.
- Valente, J. A. (1992). Logo and Special Education in Latin America. *Logo Exchange*, 10(4), 36–40.



## ISTE Books & Courseware Order Form

To order ISTE products advertised in this publication, find the product title in the following list and enter it on the form below.  
To receive a free Resource Guide with a complete listing of ISTE products and services, please call our toll-free number, 800/336-5191.

Product (* Indicates an ISTE-published title.)	Member Price	Nonmember Price
101 Ideas for Logo	19.95	19.95
Children's Machine—Rethinking School in the Age of the Computer, The	12.00	12.00
• Exploring Recursion With LogoWriter		
Student Book (Specify Apple II series, Mac, or MS-DOS data disk)	10.75	11.95
Teacher Manual (Specify Apple II series, Mac, or MS-DOS data disk)	10.75	11.95
• Graphing With LogoWriter	9.85	10.95
• Introduction to MicroWorlds—A Logo-Based Hypermedia Environment	25.15	27.95
• Introduction to Programming in Logo Using Logo PLUS	18.85	20.95
• Introduction to Programming in Logo Using LogoWriter, 3rd Ed.	24.25	26.95
• Introduction to Programming Using Terrapin Logo for the Macintosh	19.80	22.00
• Logo Learning—Strategies for Assessing Content and Process	12.55	13.95
• Logo PLUS for Educators—A Problem Solving Approach	11.70	13.00
• LogoWriter for Educators—A Problem Solving Approach	11.70	13.00
• Math Activities Using LogoWriter (Specify Apple II series, Mac, or MS-DOS data disk)		
Patterns and Designs	18.85	20.95
Investigations	14.35	15.95
Numbers & Operations	16.15	17.95
High School Math	18.85	20.95
More Investigations	14.35	15.95
More Patterns and Designs	16.15	17.95
Probability and Statistics	12.55	13.95
Media & You	29.95	29.95
• More Turtle Power—Text, Graphics, and Sound With LogoWriter (Specify Apple II series, Mac, or MS-DOS data disk)	18.85	20.95
• Turtle Power—Beginning Graphics With LogoWriter, Rev. Ed. (Specify Apple II series, Mac, or MS-DOS data disk)	20.65	22.95
• Turtle Time Designs—Working With Patterns in LogoWriter (Specify Apple II series, Mac, or MS-DOS data disk)	11.65	12.95
• Wacky Words for Traveling Tales—Logo Applications in Language Arts	15.25	16.95

**Receive an additional 18% discount when ordering 10 or more of the same title of ISTE-published products.**

Name		Membership #
School/Business		
Address		
City	State	Zip/Postal Code
Country	Phone	

## Shipping & Handling

\$0-\$15.99 (subtotal) .....	add \$4.50
\$16-\$45.99 (subtotal) .....	\$6.00
\$46-\$75.99 (subtotal) .....	\$7.00
\$76-\$100.99 (subtotal) .....	\$8.00
\$101 or more .....	8% of subtotal

**GST Registration Number 12882431**

## Shipping & Handling

\$0-\$15.99 (subtotal) .....	add \$4.50
\$16-\$45.99 (subtotal) .....	\$6.00
\$46-\$75.99 (subtotal) .....	\$7.00
\$76-\$100.99 (subtotal) .....	\$8.00
\$101 or more .....	8% of subtotal

**GST Registration Number 12882431**

## Shipping & Handling

\$0-\$15.99 (subtotal) .....	add \$4.50
\$16-\$45.99 (subtotal) .....	\$6.00
\$46-\$75.99 (subtotal) .....	\$7.00
\$76-\$100.99 (subtotal) .....	\$8.00
\$101 or more .....	8% of subtotal

**GST Registration Number 12882431**

## Shipping & Handling

\$0-\$15.99 (subtotal) .....	add \$4.50
\$16-\$45.99 (subtotal) .....	\$6.00
\$46-\$75.99 (subtotal) .....	\$7.00
\$76-\$100.99 (subtotal) .....	\$8.00
\$101 or more .....	8% of subtotal

**GST Registration Number 12882431**

## ORDER

[illegible]

## PAYMENT OPTIONS

- ☐ **Payment enclosed.** Make checks payable to ISTE—  
International orders must be prepaid with U.S. funds or credit card.  
☐ VISA    ☐ MasterCard    ☐ Discover Card    Expiration Date \_\_\_\_\_  

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
- ☐ **Purchase Order enclosed.** Please add \$4.00 for order processing—  
P.O. not including \$4.00 fee will be returned.
- ☐ **C.O.D.** for U.S. book orders only. You will pay UPS the total upon delivery by check or cash—  
ISTE will add \$4.50 order processing.
- ☐ **Airmail.** International orders for Books & Courseware are sent surface mail—  
ISTE will bill you the additional shipping charge for airmail.
- ☐ Send me ISTE membership and subscription information.
- ☐ Send me a free ISTE catalog.

Deduct 18% on ISTE-published titles if ordering quantities of 10 or more of the same title

Deduct 18% on ISTE-published titles if ordering quantities of 10 or more of the same title

Deduct 18% on ISTE-published titles if ordering quantities of 10 or more of the same title. SUBTOTAL

Deduct 18% on ISTE-published titles if ordering quantities of 10 or more of the same title.

SUBTOTAL

Shipping and Handling (see box above)

Deduct 18% on ISTE-published titles if ordering quantities of 10 or more of the same title

---

SUBTOTAL

Shipping and Handling (see box above)

Add Additional 5% of SUBTOTAL if shipped to PO Box, AK, HI, or outside U.S.

Deduct 18% on ISTE-published titles if ordering quantities of 10 or more of the same title

**SUBTOTAL**

Shipping and Handling (see box above)

Add Additional 5% of SUBTOTAL if shipped to PO Box, AK, HI, or outside U.S.

Add 7% of SUBTOTAL for GST if shipped to Canada

Deduct 18% on ISTE-published titles if ordering quantities of 10 or more of the same title

**SUBTOTAL**

Shipping and Handling (see box above)

Add Additional 5% of SUBTOTAL if shipped to PO Box, AK, HI, or outside U.S.

Add 7% of SUBTOTAL for GST if shipped to Canada

If billed with purchase order, add \$4.00; If COD, add \$4.50

Deduct 18% on ISTE-published titles if ordering quantities of 10 or more of the same title

**SUBTOTAL**

Shipping and Handling (see box above)

Add Additional 5% of SUBTOTAL if shipped to PO Box, AK, HI, or outside U.S.

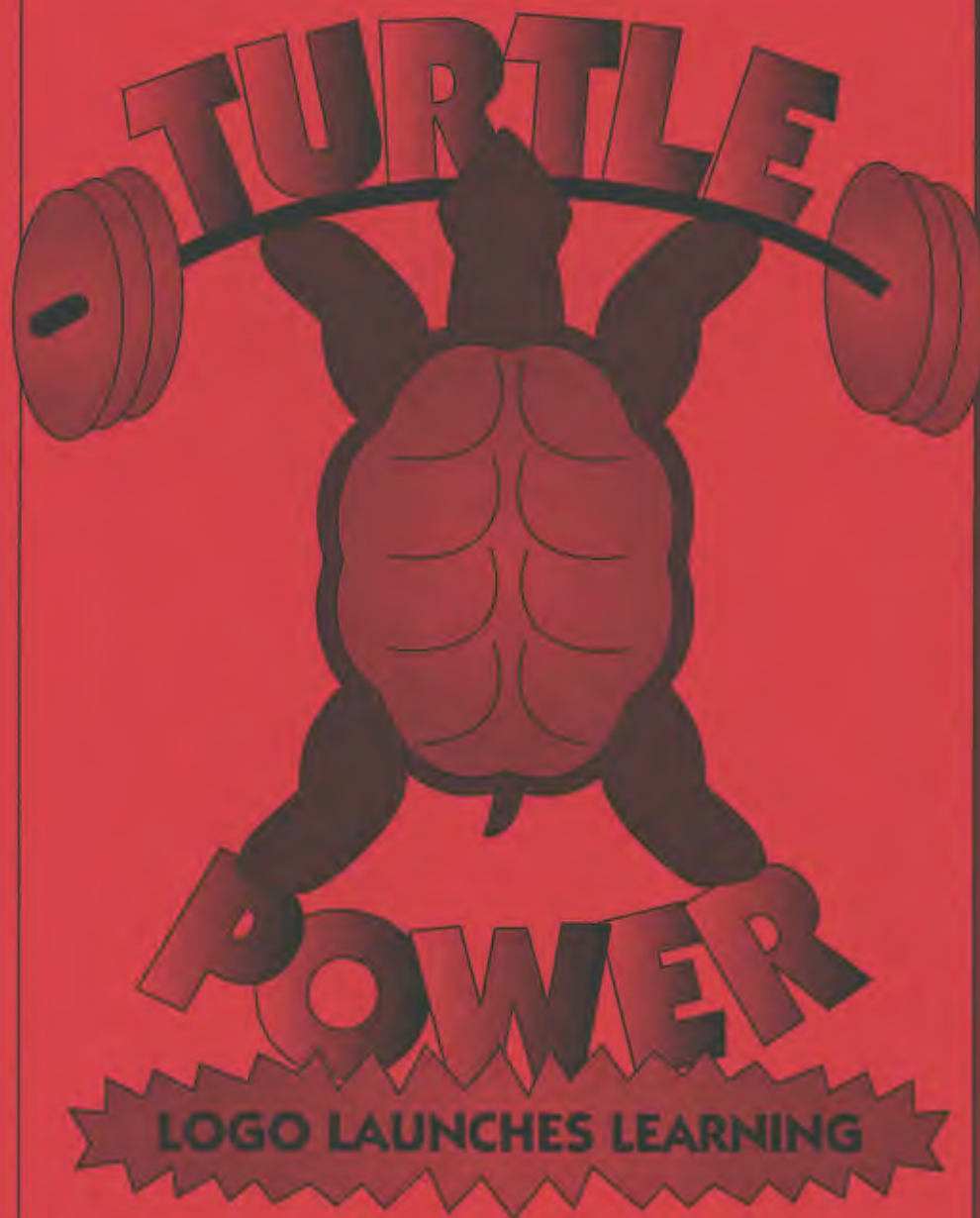
Add 7% of SUBTOTAL for GST if shipped to Canada

If billed with purchase order, add \$4.00; If COD, add \$4.50

**TOTAL**

**ISTE • 480 Charnelton Street, Eugene, OR 97401-2626 USA • Order Desk 800/336-5191 • Fax 541/302-3778\***

(\*Effective November 1, 1995, our area code changed to 541)



Many educators around the globe have been using turtle power in their classrooms with very positive results. As a beginning programming language, Logo is your logical choice.

ISTE's Logo in the Classroom series gives teachers fun, easy, and innovative methods for using Logo programming to improve the thinking and problem-solving skills of precollege students.

Logo concepts and tools can be used to teach a variety of subjects. The series offers interactive ways to use Logo for teaching mathematics, language arts, art, modeling and simulations, and computer science.



**International Society for Technology in Education**  
*Customer Service Office*  
480 Chambliss Street, Eugene, OR 97401-2626 USA  
Phone: 800/336-5191 (US and Canada)  
Phone: 541/302-3777 (International)  
Fax: 541/302-3778 Fax-on-Demand: 800/891-0212  
America Online: ISTE Compuserve: 70014,2117  
Internet: iste@oregon.oregon.edu  
World Wide Web: <http://isteonline.oregon.edu>

**The ISTE Logo in the Classroom Books include the following:**

**181 Ideas for Logo**

**The Children's Machine—  
Rethinking School in the  
Age of the Computer**

**Exploring Recursion With LogoWriter**

**Graphing With LogoWriter**

**Introduction to MicroWorlds—  
A Logo-Based Hypermedia  
Environment**

**Introduction to Programming in Logo  
Using LogoWriter, Third Edition**

**Introduction to Programming in Logo  
Using Logo PLUS**

**Introduction to Programming  
Using Terrapin Logo for the  
Macintosh**

**LogoWriter for Educators—  
A Problem Solving Approach**

**Logo PLUS for Educators—  
A Problem Solving Approach**

**Logo Learning—  
Strategies for Assessing Content and  
Process**

**Math Activities Using LogoWriter  
Series:**

**High School Math  
Investigations  
More Investigations  
Numbers and Operations  
Patterns and Designs  
More Patterns and Designs  
Probability and Statistics**

**Turtle Power—Beginning Graphics  
With LogoWriter, Revised Edition**

**More Turtle Power—Text, Graphics,  
and Sound With LogoWriter**

**Turtle Time Designs—  
Working With Patterns in LogoWriter**

**Wacky Words for Traveling Tales—  
Logo Applications in Language Arts**

## ISTE BRINGS THE WORLD OF TECHNOLOGY CLOSER TO YOU.

By drawing from the resources of committed professionals worldwide, ISTE provides support that helps educators like yourself prepare for the future of education.

ISTE members benefit from the wide variety of publications, specialized courseware, and professional organizations available to them.

They also enjoy exciting conferences, global peer networking, and mind-expanding In graduate level distance education courses.

So if you're interested in the education of tomorrow, call us today.

AL SOCIETY FOR



International Society for Technology in Education  
Customer Service Office  
480 Charndon Street, Eugene, OR 97401-2626 USA  
Phone: 800/336-5181 (US and Canada)  
Phone: 541/302-3777 (International)  
Fax: 541/302-3778 Fax-on-Demand: 800/891-0212  
America Online: ISTE Compuserve: 70014,2117  
Internet: [iste@oregon.oregonnet.edu](mailto:iste@oregon.oregonnet.edu)  
World Wide Web: <http://www.iste.oregonnet.edu>

WE'LL PUT YOU IN TOUCH WITH THE WORLD.