



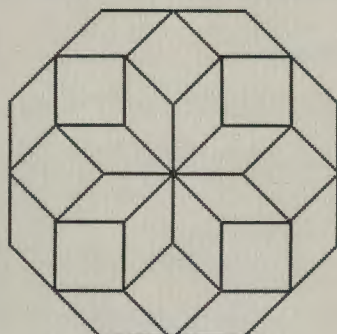
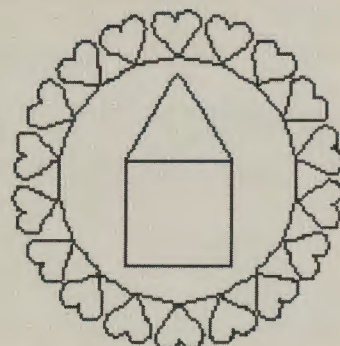
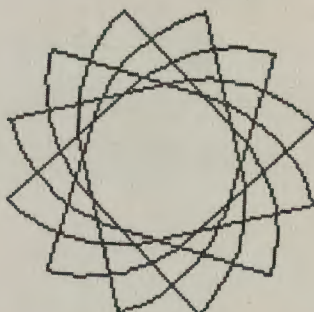
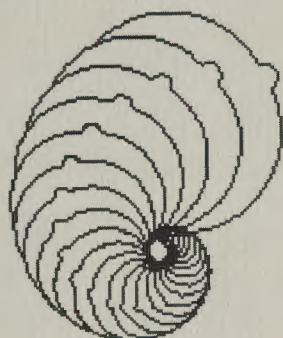
# LOGO EXCHANGE

*The Magazine for LOGO Activities Worldwide*

ISSN 0888-6970

VOLUME 5 NUMBER 4

DECEMBER 1986



Bulk Rate  
U.S. Postage Paid  
Permit #26  
Worcester, MA

Meckler Publishing Corporation

# ON LOGO

*...new video tapes for teacher training*

created by **Seymour Papert**



## **New Mindstorms (4 tapes...2 plus hours)**

- . concentrates on learning and thinking
- . introduces Logo to the beginner
- . provides other educator and parent groups with a basis for understanding

## **Logo Hurdles (4 tapes...2 plus hours)**

- . teaches aspects of Logo that are often difficult to understand and master
- . can be used by individuals or groups

*Either series with print and diskettes-\$699*

*PAL version \$999*

***SPECIAL OFFER, BOTH SERIES PLUS 2 FREE  
LOGOWRITER TAPES, \$1295***

**mmi** 12 clayton terrace saint louis, missouri 63131



# LOGO EXCHANGE

*The Magazine for LOGO Activities Worldwide*

ISSN 0888-6970

VOLUME 5 NUMBER 4

DECEMBER 1986

## Editor-in-Chief

Tom Lough

## International Editor

Dennis Harper

## Assistant Editors

Glen Bull

Steve Tipps

## Field Editors

Eduardo Cavallo

Patricia Dowling

Michael Friendly

Anne McDougall

Richard Noss

Fatimata Seye Sylla

## Senior Contributing Editor

Robb Muir

## Contributing Editors

Elaine Blitman

Regina Bowden

Doug Clements

Paula Cochran

Bill Craig

A. J. (Sandy) Dawson

Barbara Elias

Anne Cairnes Federlein

James Fry

Judi Harris

Barbara Jamile

Jim McCauley

Rebecca Poplin

Barbara Randolph

Linda Sherman

## Managing Editor

Anthony Abbott

## Advertising Director

Jean-Paul Emard

## Publisher

Alan M. Meckler

*Logo Exchange* (ISSN 0888-6970) combines *National Logo Exchange* and *International Logo Exchange* and is published monthly September through May by Meckler Publishing Corporation, 11 Ferry Lane West, Westport, CT 06880, (203) 226-6967.

Copyright © 1986 by Meckler Publishing Corporation. All rights reserved. No part of this publication may be reproduced in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage retrieval system, without permission in writing from the publisher.

Application to mail at Second Class postage rates is pending in Westport, CT and additional mailing offices. POSTMASTER: Send all address changes to *Logo Exchange*, Meckler Publishing Corporation, 11 Ferry Lane West, Westport, CT 06880.

## CONTENTS

From the Editor	2
Trim a Logo Christmas Tree	3
Tipps for Teachers	4
Teacher Feature	5
Teaching Tools	7
TO BEGIN:PROJECTS	10
Logo LinX	12
Math Worlds	14
SpecialTalk	18
Testudinal Testimony	21
Logo Disserts	23
Teacher to Teacher	24
LXionary	25
The Adventures of Jacques and Elsie	26
Q and A	27
IntLXual Challenges	28

## International Logo News

Global Comments	29
Europe	30
Asia	31
Latin America	32

**Cover:** The Logo "poinsettia" design was one of the ornaments created by Edwina Walsh's seventh grade students for their Christmas tree project. Edwina teaches in LaPalma, California.

Subscription: \$29.95 per year. Subscribers outside the USA please add \$6.00 for surface mail or \$15.00 for airmail. Orders from North and South America should be sent to *Logo Exchange*, Dept. VV, P.O. Box 3000, Denville, NJ 07834; elsewhere to Meckler Publishing, 3 Henrietta Street, London WC2E 8LU, UK, (tel.) 01-240-0856.

## From the Editor

by Tom Lough

### Upstart STARTUP

"R-r-r-r-r. R-r-r-r-r."

That's the sound our old Buick station wagon makes when I try to get it started on a winter morning. It usually takes a couple of tries, but soon the engine roars into life, billowing out clouds of condensation from the tailpipe. After a few more minutes to warm up, the wagon is ready for the day.

One morning recently, while scraping the frost off the windshield as the engine warmed up, I began thinking about the process I went through to get the car started. Quite naturally, my thoughts soon turned to the STARTUP variable of Logo.

STARTUP is a variable of power. It enables one to configure a Logo file [or even a language disk] to run a procedure automatically, making it possible to produce "turnkey" Logo-based programs.

I certainly wished that there was a STARTUP variable available for my old Buick! All I would have to do would be to put in the key [disk] and turn it [switch it on]. An instruction something like MAKE "STARTUP [BEGIN] would have been executed previously. The car [computer] would look for the STARTUP variable, and then would find and run the procedure called BEGIN. I'd be on my way quickly, with no more waiting for "R-r-r-r-r."

Then I got to thinking about STARTUP in general. In my teaching, sometimes it is difficult to get a new activity started, particularly if it is something which the students have

never done before. When it is something which I have not done before, the task seems even more difficult!

How often do we become frustrated or discouraged by the extra time and effort which accompanies the normal "startup" process of an activity?

Experiences such as these have helped me to recognize and accept this as a natural part of doing anything different from the usual course of events. In other words, when we attempt to climb out of the rut, when we try something different, when we respond to an invitation to grow, things sometimes get temporarily uncomfortable ... and that's normal. (How I wish there were a STARTUP variable for life!!)

By this acceptance, I have become aware of a reservoir of patience which I did not know existed. Although I still have plenty of room for improvement, I am sensing that I now have more patience with myself when I am trying something new. And, as importantly, I have more patience with my students when they finally find the courage to step out in an unfamiliar direction. Perhaps I need to think about something like MAKE STARTUP [PATIENCE].

And speaking of starting up and patience, I'd like to thank our thousands of subscribers worldwide for your patience and understanding back in September and October. Starting up the *Logo Exchange* magazine from the two newsletters had its share of difficulty. The mail system took some time to get established. I hope that things are more satisfactory now.

What new things are you thinking about? Why not use some of the holiday time for musing about these activities? And then, MAKE STARTUP [NEW.YEAR].

FD 100!



# Trim a Logo Christmas Tree

by Edwina Walsh

"There, that should do it!"

The seventh graders crowded around the computer as the final procedure was edited. Someone typed TREE and pressed <return>. You could hear the collective intake of air as everyone held their breath and watched and waited...

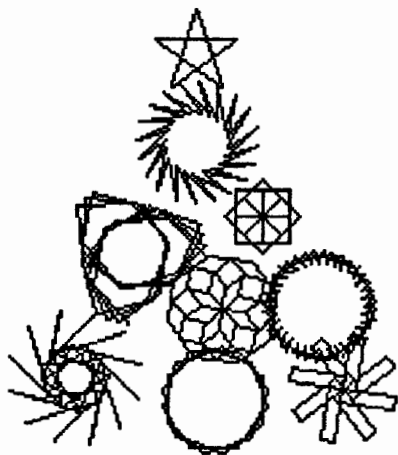
## Tannenbaum Turtle

At our school, all seventh graders take a quarter class called "Exploring Computers." Of course, we explore computers through Logo. The thirty students work two at a station; each station chooses its own projects. Last year, one of our most successful projects was a Logo Christmas tree similar in its collaborative approach to the "Logo Quilt" described in Dan and Molly Watt's article in the May 1986 NLX.

In every class, at least five or six stations chose to work together on the Christmas tree. Using variables and primitives such as SETHEADING, students created polygons that looked like stars, houses, or simply abstract tree ornaments. We decided that the design for the ornaments could be no longer than twenty turtle steps in any direction, and had to begin with the turtle's heading at zero.

Each station used its own disk. To be sure they would be easily identified, all project files were named after the station number. The students best at loading and saving files collected all the ornament procedures on a class disk. When we came to the logistics of arranging the ornaments, I taught the class SETPOS and together we created the TREE procedure that "hangs" or positions the ornaments.

It was the TREE procedure which gave an atmosphere of suspense to the classroom. When TREE ran successfully, a cheer went up, and everyone crowded close to admire their collective handiwork.



From looking at the figure, however, you cannot experience the delight in watching it grow on the screen. You cannot see its bright colors or the shining star at the top with rays which blink on and off as well as any storebought Christmas star.

## Collaborative Concerns

Although our project did not require the precise pattern of a quilt, it did need careful planning for size, heading, and symmetry. The students who arranged the ornaments had two tricky problems. First, sometimes there were inexplicable delays between ornaments. Secondly, sometimes entire ornaments would disappear!

After much discussion, debugging, and showing the turtle, they found the cause of the first problem. Several procedures had too large a number in the REPEAT command. When a procedure is run independently, it may not matter if the turtle repeats a circle one hundred times in the same place. However, such repetition causes too long a delay in a collaborative design. I was pleased that the necessity to determine exactly how many repetitions were needed to complete a design arose naturally instead of being motivated by a workbook exercise.

The second problem, disappearing ornaments, was caused by procedures that included the CLEARSCREEN primitive. If the TREE superprocedure called an ornament procedure which contained CLEARSCREEN, then all ornaments previously drawn were erased! Again, the requirement for compatible procedures arose as a natural consequence of the objective of the group project.

## A Tree-mendous Project

After participating in these exciting projects with my students, I must agree with Dan and Molly Watt. Collaborative projects stimulate interesting ideas and powerful problem solving. And, best of all, they make learning fun!

Merry Christmas, everyone!

---

*Edwina Walsh is computer coordinator and mentor teacher in computer applications at Walker Junior High School in La Palma, CA. Her current interest is student use of Logo to access laser disks.*

---

# Tipps for Teachers

by Steve Tipps

## By Any Other Name (or, How Do You Describe a Language Such as Logo?)

In Rodgers and Hammerstein's *The Sound of Music*, the Mother Superior wondered how to describe a problem like Maria. Her suggestions included "a flibberti gibbet, a will-o'-the-wisp, a clown." At meetings and conferences, presenters have often had a similar problem when attempting to describe the qualities which make Logo so appealing. And they have also often been moved to metaphor. In the last five years of learning with Logo, a large number and wide variety of descriptions have been used for our favorite computer language.

### Logo Is a Room

"No threshold, no ceiling" is possibly the first Logo description encountered. By this statement, people mean that Logo as a language requires no special background to enter, but having entered in this Logo room, house, or building, learners find no predetermined limits in the language. This infinite Logo space offers unlimited possibilities which we can explore. Perhaps, the space is a room without boundaries, learning without limits, a school without walls.

Experience, however, suggests that learners do find obstacles in the Logo room. Most learners have hit, at various times, what is called a Logo wall. Some Logo learners encounter an obstacle with the concept of variables.

Variables, useful and essential, seem to be a genuine thinking obstacle for some people. The punctuation problem of quotes (") and dots (:) compounds the issue. However, the real problem is a conceptual one which is probably also the problem people have with algebra.

State transparency is another obstacle which people often find. I once observed a group of students who had SQUARE defined in a non-transparent manner.

```
TO SQUARE
FORWARD 50 RIGHT 90
FORWARD 50 RIGHT 90
FORWARD 50 RIGHT 90
FORWARD 50
END
```

The SQUARE drew a square well enough, but the turtle did not return to its original place and direction. The turtle was an undependable thinking object; each new square did something a little unexpected because it started out in the "wrong" direction.

You no doubt have observed and have your own history of Logo walls. My major thinking problem has been recursion used "correctly" rather than as a loop. Simple tail recursion is understandable on its face; even programs which demonstrate the outputs popping back through the duplicates are reasonably sensible to me. However, programs which actually use recursion to do anything more than the most trivial of tasks baffle me.

The gap between the "no threshold" of turtle graphics and the "no ceiling" of list processing has also led some observers to wonder if Logo also has "no middle." Thinking with lists would appear to be an everyday situation with mailing lists, shopping lists, and "to do" lists. But manipulation of lists in real life does not appear to prepare people for manipulation of Logo lists. A major philosophical difference between the dialects of Logo is how much of the list structure to expose during turtle graphics. The difference between SETPOS [ 40 30 ] and SETXY 40 30 is much deeper than the brackets.

The Logo room is actually a room filled with walls which different learners encounter at different times and in different ways. Instead of an infinitely open room with no limitations, Logo is a room filled with interesting mazes. You get to a dead end when you run into another Logo wall which stymies and frustrates you. A little time is needed to backtrack and to figure out which turn was missed.

### Logo Is a Toolbox

Many people see Logo as a toolbox. The Logo Toolbox provides many different tools for many different tasks. When you have a new job, just look into the tool chest and see what is available.

Modularity is part of the tool metaphor. Each procedure is a tool which does its own job and can also be used in other ways. In fact, if you don't have a tool ready made for the task, use the tools which are available to make a new tool. I have had to learn and relearn this many times.

I remember complaining because Terrapin Logo had no WAIT command and LCS Logo had no NODRAW. What a revelation when I understood I could write those commands for myself! I began to think differently about the language. Logo could do more than draw squares. It could also print procedures in order and help students explore standard deviation.

The realization of extensibility is fundamental in Logo and is an attitude which I want in all computer use. After working with Logo, students are more demanding of the word processor and spreadsheet and less willing to accept apparent limitations. When students say "There must be a better way" or "I don't like the way this looks, I want it to...", they are seeing the computer as a tool to do their bidding.

Other Logophiles describe Logo as a specific tool. Logo is a shovel; you must look for fertile place to dig. A more humorous and ominous metaphor has Logo being a chain saw. Logo is very powerful, but you better watch out what you aim at. Some utilitarian views of Logo see Logo as an erector set or construction set.



Whether Logo is described as a toolbox or as a specific apparatus, people have tried to convey the sense of Logo as multipurpose, flexible, and personal. Enablement is the key that opens the Logo toolbox. James Burke, who narrated a wonderful series for BBC and PBS called *Connections*, recently lectured here. He made the point that while people work with tools, tools also change how people work.

### Logo Is a Sandbox

The sandbox view has Logo being malleable and squeeze responding to the creative desires of the player. This playful metaphor retains the highly personal emphasis of the toolbox metaphor and adds an affective dimension. Fingerprinting is another metaphor which alludes to the same qualities.

Piaget once defined play as pure assimilation. The learner is no longer struggling to work something out but uses new knowledge in imaginative ways. The creative, playful side of Logo is apparent to anyone who has watched children (or adults) spin a square around and squeal with glee. The new combination of knowledge of the SQUARE and of rotation using a REPEAT is play. The learning, adaptation in Piagetian terms, may have been rapid or cautious, but the joy of making the square whirl around can be repeated 10 or 20 times. It is an affirmation of understanding, the joy of learning.

The turtle, Logo's own icon, belongs in this playful, joyful category. The turtle represents the friendliness and unconditional acceptance by Logo. Mistakes are okay and expected. Thinking like the turtle provides a vehicle for metacognition. When we say that the turtle is turtlecentric, we provide a way for students to move from egocentrism. The turtle captures the imagination and enables a different cognition for students.

### Metaphorically Speaking

Logo, in addition to being a language, is itself a metaphor for a new way of thinking and learning. Papert said as much when relating his own learning with gears. Gears were his thinking device.

Logo may be a room, a maze, a toolbox, a shovel, a construction set, or a sandbox. Some other metaphor may best describe your feelings and relationship with Logo. At times, I think Logo is a snarling dog (or a snapping turtle) which has me cornered and won't let go. By any of its other names, Logo is just as exciting, as useful, as playful, or as frustrating.

If you have favorite metaphors for Logo, please share them with me. Send your metaphor along with a brief discussion to:

Steve Tipps  
Midwestern State University  
Wichita Falls, TX 76308

---

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX, and has been involved with Logo since 1982. He conducts Logo workshops for school systems throughout the United States, and is a popular conference speaker. His CompuServe number is 76606,1623.

## Teacher Feature

by Rebecca Poplin

### Featuring: Gwen Roberts

#### Lights in Their Eyes

It was the lights in the eyes of children that inspired Gwen Roberts to begin teaching. When Gwen found that Logo was a source of power for those lights, she became a Logo fanatic.

Gwen came to Logo by a circuitous route. Having left Oklahoma to go to medical school in California, Gwen has been a dancer, a waitress, and a play producer. In the sixties, Gwen taught in a free school classroom which was truly portable. It was a school bus! When the students weren't taking field trips, the bus served as a math classroom.

Gwen frequently tutored in math, and there she discovered her enchantment with the lights of understanding.

#### It Happened After School

In 1983, Gwen worked in an after-school tutoring program at the neighborhood public school her children attended. Dave Mintz was teaching Logo to a group of peer trainers while Gwen happened to be there. When a child got up to go to the bathroom, Gwen would be in the vacated chair, fascinated with her own exploration of turtle graphics.

Dave heard of a job opening at The Open School in Los Angeles and suggested to Gwen that she apply. Since then, she has been learning with her students and becoming more involved with Logo and other computer applications.

#### The Open Opportunity

The Open School is a magnet school in Los Angeles, serving 240 students in bungalows on the campus of another school. The school population is balanced ethnically and economically, and serves both high and low achievers.

Gwen works in a computer lab with eight Macintosh computers. She is using LCSi Logo for the Macintosh, which has word processing capability. Gwen sees every child in school about once a week for thirty minutes.

Sometimes she goes into their classrooms, in which an additional five to ten Macintoshes serve each group of sixty children and two teachers.

The Open School program is primarily individualized with learning centers. The plan is that the computers should become as invisible as pencils and paper.

Gwen uses the lab during lunch and recess for really free time. Students can come in with their own ideas or just a burning need to do something with the computer. There is usually a full house!

### Sines of Ancient Times

One of Gwen's favorite projects last year began with using similar triangles to measure distances like the height of the corner of the room. Students measured all sorts of things you can't reach and developed the idea of triangulation for measurement. They made simple astrolabes with copies of protractors to which they glued straw and string. The next step was to compare the ratio of the sides for all the different angles they had created. They used a simple Logo procedure to do their comparisons.

Gwen had encouraged the use of 30 degree angles, so many of the fractions turned out equivalent to one half! Gwen and the students were amazed together about their discovery. Soon, Gwen brought to class the "funkiest" old musty math books she could find. The children were amazed that their discovery was actually in a sine table in those books.

Logo inspired Gwen and the children to wonder why 360 is the number of degrees in a circle. It is a strange number, after all. They discovered that in Babylonia, the ancients studied the heavens intently and found that it took about 360 days for the night sky to look the same again. Even though they thought the sky rotated around the earth, the principles

were important. Gwen and her students decided that it is lucky for us that they estimated a little incorrectly, or our calculations would be much more difficult. In each of these projects, Logo was a catalyst for rediscovering ancient mathematical knowledge.

At the West Coast Logo Conference in March, Gwen will be sharing more details about her Logo class work, and some plans for an interesting Logo project.

Meanwhile, if you are interested in learning more about The Open School or Gwen's projects, write to:

Gwen Roberts  
526 South Hudson Avenue  
Los Angeles, California 90020

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

Rebecca L. Poplin  
2421 Fain Street  
Wichita Falls, Texas 76308

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

## Key Reference Sources

### LOGO EXCHANGE RESEARCH DIRECTORY

Edited by Regina H. Sapona

Price: \$9.95 (paper)

ISBN: 0-88736-085-8 (1986)

### NATIONAL LOGO EXCHANGE INDEX (volumes 1-4 (1982-1986))

Compiled by Regina H. Sapona

Price: \$9.95 (paper)

ISBN: 0-88736-090-4

### Order Form

Please enter orders for the following:

- ☐ LOGO EXCHANGE RESEARCH DIRECTORY @ \$9.95  
☐ NATIONAL LOGO EXCHANGE INDEX (1982-1986) @ \$9.95

☐ Payment enclosed (required for all personal orders, which must be prepaid by personal check or credit card)

Name \_\_\_\_\_  
 Organization \_\_\_\_\_  
 Address \_\_\_\_\_  
 City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

☐ Please bill my organization Purchase Order No. \_\_\_\_\_  
 Authorized Signature \_\_\_\_\_



Meckler Publishing  
Corporation  
11 Ferry Lane West  
Westport, CT 06880  
203-226-6967

☐ Credit card payment (circle one):

AMEX      MasterCard  
 VISA      Diners Club

Card No. \_\_\_\_\_

Exp. Date \_\_\_\_\_

Cardholder's Signature \_\_\_\_\_



# Teaching Tools

by Glen Bull and Paula Cochran

## Transformations: Advanced Geoboard Tools

In last month's column, we described several Logo tools which can be used to enhance explorations with geoboards. A geoboard is a wooden board with nails or pegs. Rubber bands are stretched around the pegs to form geometric figures. Each of the pegs has an X (horizontal) and Y (vertical) coordinate. Thus it is possible to describe the location of any peg with two numbers which represent the X and Y coordinates, respectively.

GRAPH is a tool we created for graphing a similar lists of coordinates with Logo. The purpose of the tool is not to replace work with the geoboard itself, but to enhance it. This tool consists of two procedures: the master procedure called GRAPH, and a subprocedure GRAPH.WORK which does all of the work.

If you are using Terrapin Logo, omit the brackets around [STOP] in GRAPH.WORK. You will also need to add the SETPOS procedure found at the end of the column.

```
TO GRAPH :LIST
  PU SETPOS FIRST :LIST
  PD GRAPH.WORK BF :LIST
END
```

```
TO GRAPH.WORK :LIST
  IF EMPTY? :LIST [ STOP ]
  SETPOS FIRST :LIST
  GRAPH.WORK BF :LIST
END
```

To use GRAPH, provide a list of X and Y coordinates which represent a figure.

```
? GRAPH [ [ 5 0] [0 10] [15 10] [15 30]
[30 15] [20 15] [20 10] [35 10] [30 0] [5 0] ]
```

The figure of a sail boat has ten points which must be specified. This represents quite a bit of typing. However, after you are satisfied with a particular design, the list of coordinates can be put in the form of a procedure. This eliminates the necessity of retyping the list every time it is used.



```
TO BOAT.LIST
  OUTPUT [ [ 5 0] [0 10] [15 10] [15 30] [30 15] [20 15]
[20 10] [35 10] [30 0] [5 0] ]
END
```

To produce the same graph as before, type this:



```
? GRAPH BOAT.LIST
```

The coordinate numbers can be printed as well as graphed.

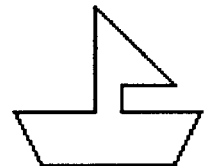
```
? PRINT BOAT.LIST
[ 5 0] [0 10] [15 10] [15 30] [30 15] [20 15] [20 10]
[35 10] [30 0] [5 0]
```

### Transformations

The boat that we have drawn is a rather small one. In the last issue we provided three tools to enlarge figures: MAGNIFY.X, MAGNIFY.Y, and SCALE. For example, to double the scale of the boat, we would type:

```
?GRAPH SCALE 2 BOAT.LIST
```

The procedures for these tools are found at the end of the column. The SCALE procedure changes the scale of a design. In this instance, the each coordinate is doubled. Compare the two scales:



```
?PRINT BOAT.LIST
[ 5 0] [0 10] [15 10] [15 30] [30 15] [20 15] [20 10]
[35 10] [30 0] [5 0]
```

```
?PRINT SCALE 2 BOAT.LIST
[10 0] [0 20] [30 20] [30 60] [60 30] [40 30] [40 20]
[70 20] [60 0] [10 0]
```

### Shifting and Rotating

Changing the scale of the drawing is only one of the transformations which can be accomplished with the geoboard tools. The boat can also be moved to the left or the right. This is accomplished by adding to the X coordinate (or subtracting if we want to move to the left).

SHIFT.X is a procedure which can be used to add to each X coordinate. ( If you are using Terrapin Logo, omit the brackets around [ OP [ ] ] )

```
TO SHIFT.X :X :LIST
  IF EMPTY? :LIST [ OP [ ] ]
  OP FPUT ( FPUT ( ( FIRST FIRST :LIST ) + :X )
  BF FIRST :LIST ) SHIFT.X :X BF :LIST
END
```

Note: Experiments with our proofreaders have demonstrated that it is easy to make a typing mistake with these procedures, and reverse or leave out an FPUT, FIRST, or BF.

Type the procedures in carefully, and then try them out. Then give your students copies on disk. Since they are tools, it is not the procedures themselves in which we are interested, but what can be learned by using them.

If we wish to move the boat 40 steps to the right, add 40 to each X coordinate.

?GRAPH SHIFT.X 20 BOAT.LIST



To move the boat 50 steps to the left, subtract 50.



?GRAPH SHIFT.X -50 BOAT.LIST

To examine the effect of SHIFT.X on the coordinates, print the results.

? PRINT BOAT.LIST

[5 0] [0 10] [15 10] [15 30] [30 15] [20 15] [20 10]  
[35 10] [30 0] [5 0]

? PRINT SHIFT.X 20 BOAT.LIST

[25 0] [20 10] [35 10] [35 30] [50 15] [40 15] [40 10]  
[55 10] [50 0] [25 0]

A procedure to shift the boat up or down adds to (or subtracts from) the Y coordinate instead of the X coordinate. We have provided a SHIFT.Y procedure at the end of the column.

### Flipping the Design

If shifting the figure involves addition, and changing the scale involves multiplication, what transformation is required to reverse the design? The sailboat shown above has a sail on the right. What must be done to the coordinates to move the sail to the left?

In this case, making all the X coordinates negative will reverse the figure. Changing 5 to -5, 15 to -15, etc. has the effect of flipping the design horizontally. This can be accomplished by multiplying each X coordinate by -1. The MAGNIFY.X procedure from last month (reprinted at the end of the column) can do this.

```
TO FLIP.X :LIST
OP MAGNIFY.X -1 :LIST
END
```

Here is the effect that FLIP.X has on the boat:

?GRAPH FLIP.X BOAT.LIST



To flip a figure from top to bottom, rather than horizontally, make all the Y coordinates negative instead of the X coordinates.

```
TO FLIP.Y :LIST
OP MAGNIFY.Y -1 :LIST
END
```

### Concatenating Commands

It is possible to use more than one geoboard tool at a time. For example, SHIFT.X and FLIP.X can be used together. However, the order may make a difference. What is the difference in the effect of the following two commands?

```
? GRAPH SHIFT.X -60 FLIP.X BOAT.LIST
? GRAPH FLIP.X SHIFT.X -60 BOAT.LIST
```

### Programming Challenge

Can you make a procedure that will translate turtle positions into coordinates? The procedure should take a list of turtle commands such as [ FD 30 RT 45 FD 50 ] and generate a list of X and Y coordinates for each position of the turtle.

### Other Transformations

It is possible to determine a goal in advance, such as creating a tool to shift the design, and then write a Logo procedure to perform the task. It is also possible to work in exactly the opposite direction, and devise a tool which transforms the coordinates without knowing its effect in advance.

For example, what happens if we exchange all the X and Y coordinates with one another? That is, [5 0] should become [0 5], [15 10] should become [10 15], and so on. Here is a tool which does just that. The second line of the procedure (beginning "OP FPUT" and ending "BF :LIST") should be typed in as one line without pressing the RETURN key.

```
TO EXCHANGE.XY :LIST
IF EMPTY? :LIST [ OP [] ]
OP FPUT ( FPUT LAST FIRST :LIST BL FIRST
:LIST ) EXCHANGE.XY BF :LIST
END
```

You will know that you have the procedure typed in correctly when the procedure reverses the coordinates in BOAT.LIST.

```
? PRINT BOAT.LIST
[5 0] [0 10] [15 10] [15 30] [30 15] [20 15] [20 10]
[35 10] [30 0] [5 0]
```

**? PRINT EXCHANGE.XY BOAT.LIST**

[0 5] [10 0] [10 15] [30 15] [15 30] [15 20] [10 20]  
[10 35] [0 30] [0 5]

What is the effect of using EXCHANGE.XY?

**? GRAPH EXCHANGE.XY BOAT.LIST**

Were you as surprised as we were when we first tried this procedure? Can you think of a better name for the procedure than EXCHANGE.XY?

**Geoboards Revisited**

It is important not to use Logo as a substitute for other media. It should be a tool which enhances rather than replaces them. It is no accident that we designed our geoboard tools so that the effects of a transformation could be observed using the PRINT statement.

Challenge students to think about the problems using rubber bands on geoboards and lists of numbers written with pencil and paper before sending them to the computer. You should also encourage cross-fertilization in both directions. For example, the scale of a design on the computer can be increased so that it becomes a mural for the cafeteria or for a school dance.

A comparison of the SCALE procedure ( and its subprocedures MAGNIFY.X and MAGNIFY.Y ) with SHIFT.X and SHIFT.Y is worthwhile. Changing the scale of the design involves multiplying each coordinate. Moving the figure up or down involves addition of a set amount to each coordinate. The same ideas pop up in other mathematical areas. For example, in the algebraic formula  $Y = AX + B$ , the variable "A" involves multiplication, and changes the slope (scale) of the graph, while the variable "B" is added, and moves the overall graph up or down.

If you have access to MacPaint or MacDraw, or their equivalents on the Apple or the IBM ( Mousepaint, PC-Paint, etc. ), encourage students to develop designs and use the paint tools to rotate, flip, or transform them. Then discuss the underlying mathematics that must be used to accomplish these tasks.

A great deal can be learned about coordinate geometry and algebraic plots by experimenting with these tools. However, with experimentation it becomes clear why Seymour Papert developed turtle geometry as a first introduction to computer math. The purpose, as always, lies in the process rather than the product.

---

*Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. His CompuServe number is 72477,1637. Paula Cochran is an assistant professor in the Communication Disorders Program of the University of Virginia's Curry School of Education. She is interested in Logo applications for language arts and special populations.*

**Tools Reprinted from Last Month**

The magnification tools are used to change the scale of a figure. The MAGNIFY.Y procedure stretches the height of a figure. MAGNIFY.X changes the width. SCALE changes both dimensions at once. (In Terrapin Logo, it is necessary to omit the brackets around [ OP [ ] ].)

```
TO MAGNIFY.X :X :LIST
  IF EMPTY? :LIST [ OP [ ] ]
  OP FPUT ( FPUT ( ( FIRST FIRST
:LIST ) * :X ) BF FIRST :LIST ) MAGNIFY.X
: X BF :LIST
END
```

```
TO MAGNIFY.Y :Y :LIST
  IF EMPTY? :LIST [ OP [ ] ]
  OP FPUT ( LPUT ( ( LAST FIRST
:LIST ) * :Y ) BL FIRST :LIST ) MAGNIFY.Y
: Y BF :LIST
END
```

```
TO SCALE :SIZE :LIST
  OP MAGNIFY.X :SIZE MAGNIFY.Y
:SIZE :LIST
END
```

**Shifting the Turtle**

The companion procedure for SHIFT.X (provided in the body of the column) is SHIFT.Y. The SHIFT.Y procedure makes it possible to move a design up or down.

```
TO SHIFT.Y :Y :LIST
  IF EMPTY? :LIST [ OP [ ] ]
  OP FPUT ( LPUT ( ( LAST FIRST
:LIST ) + :Y ) BL FIRST :LIST ) SHIFT.Y
: Y BF :LIST
END
```

**Procedures Required for Terrapin Logo**

If you are using Terrapin Logo, you will need to type in the following SETPOS procedure.

```
TO SETPOS :LOCATION
  SETXY FIRST :LOCATION LAST
:LOCATION
END
```

If you are using Version 1 of Terrapin Logo, you will also need an EMPTY? procedure.

```
TO EMPTY? :LIST
  OP :LIST = [ ]
END
```

## TO BEGIN :PROJECTS

*by Elaine Blitman and Barbara Jamile*

Teaching Logo through a project-centered approach allows children to use the computer as a tool in a content area while learning Logo skills. This keeps the focus of learning on content objectives, and demonstrates that computers offer another way to learn. Children learn skills as they are needed in their projects and remember them as part of a meaningful whole.

As soon as some basic commands for moving the turtle are learned, children are ready to begin working on projects. Complex projects usually take several sessions to complete, thus students will need to save their work as procedures on a disk. Let's take a look at a sample third grade project assignment using Apple Logo. (Other versions of Logo may be used in a similar way, although commands may vary slightly.)

For example, an awareness of the meaning of such traditional holiday symbols as bells, pine trees, menorahs, and yule logs is a social studies objective. One activity in the unit calls for every child to select a symbol for study. Students may use the computer for their presentation.

Some children may begin by creating a picture of the symbol on the computer, while others research the meaning of the symbol. These projects have a "title page" naming the symbol, the author, and the date. Next comes a picture of the symbol, identified by a title, a sentence or two of text, or perhaps even a poem. The text is added by using a splitscreen, which allows four lines of print at the bottom of the graphics screen. After the graphic is cleared, a paragraph about the symbol appears on the full textscreen. Music (Terrapin / Commodore / LogoWriter / IBM) can be added by more advanced users. The degree of complexity is determined by the student.

The title page uses SETCURSOR, TEXTSCREEN, and WAIT. Paper copies of textscreen grids (40 columns and 24 rows) help the students plan the page. The procedure can also be done on paper before going to the computer. The WAIT time between pages is usually determined on the computer after the procedure is entered.

Up to four lines of text can appear below the graphic when using a splitscreen. If you want all four lines of text to appear on the screen, it is important to use TYPE in place of PRINT. This keeps the cursor at the end of the last line rather than at the beginning of a new line (thus scrolling up the first line of text.)

The Logo editor can be used like a word processor. Use PRINT commands within a procedure to write a paragraph.

This way, it can be called up by a superprocedure as a page in the project.

```
TO STORY
PR [ HOLLY'S GLOSSY GREEN LEAVES ]
PR [ AND BRIGHT RED BERRIES ]
PR [ ARE PERFECT FOR HOLIDAY WREATHS. ]
```

```
PR [ LONG AGO HOLLY WAS USED ]
PR [ IN HOUSES AND CHURCHES ]
PR [ AT CHRISTMAS TIME AND ]
PR [ WAS CALLED "HOLY" TREE. ]
END
```

Logo skill building takes place as these projects are completed. Saving and retrieving on a disk, renaming a file, editing, recycling for added space, and new graphics commands are explored and explained as needed. Some techniques are appropriate for the entire group while others may be directed to individuals for specific problem solving.

Each child presents his or her completed project to the group, using the computer. It is helpful to have two computers available during presentations so that one child may load a program while another presents. Parents also enjoy these show and tell sessions. Printouts of the projects can be displayed or sent home. An important outcome is achievement of the social studies objective, while also solving many Logo problems.

Once students have experienced the process of designing a project, presenting information on the computer becomes a regular option in other areas, such as book sharing or science reports.

### Star-Finding

"Imagination is the star in man." --C. G. Jung

This project fits the mood of the holiday season. The methods utilize the special features of LogoWriter, but can easily be modified for other versions of Logo. Exploration begins with the commands that form a five-pointed star.

```
?REPEAT 5 [ FD 50 RT 144 ]
```

How might other stars be formed?

Children can experiment with random angles, keep records and pass around "good angles for making stars". A five-pointed star can also be made by using the 360 degree rule and adding the multiplier 2 at the end:

```
?REPEAT 5 [ FD 50 RT 360 / 5 * 2 ]
```

Use the computer to figure out the angle:

```
?SHOW 360 / 5 * 2
144
```

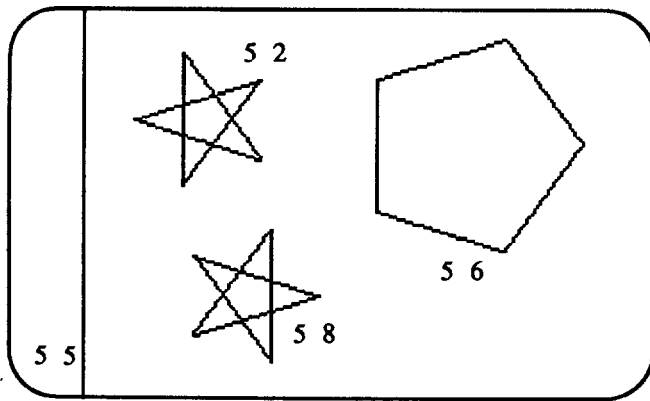
What happens when the multiplier isn't used? Try:

```
?REPEAT 5 [ FD 50 RT 360 / 5 ]
```

Edit the REPEAT command to change only the multiplier to some other values.

```
?REPEAT 5 [ FD 50 RT 360 / 5 * __ ]
?SHOW 360 / 5 * __
```

A sequence of numbers can easily be tried by using the LogoWriter UP arrow and DELETE key. Make a visual record by using APPLE 8 (LABEL) after each figure is drawn; type the number of sides and the multiplying number beside it. Find another open space on the page to draw the next figure by using the TURTLE MOVE option. As soon as the page is complete, use PRINTSCREEN to make a printed star record.

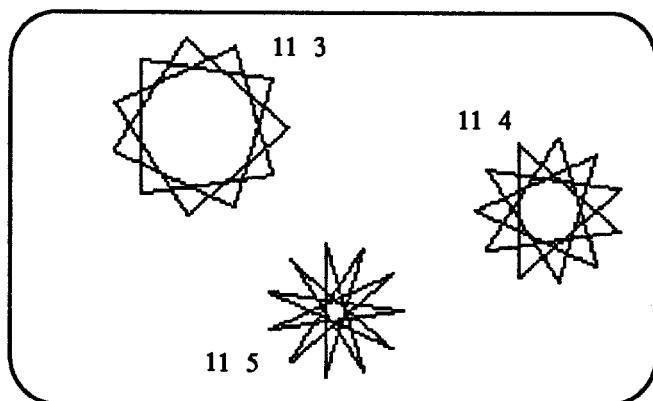


See if the children can predict what the next figure will be after examining the collection of shapes. After several combinations with the number 5 have been tried, edit the procedure to change the number.

Students who are ready to try variables might enjoy having this tool for finding stars:

```
TO FIND :SIDES :NUM
REPEAT :SIDES [ FD 50 RT 360 / :SIDES * :NUM ]
END
```

Type FIND 11 2 ( or the numbers you choose ) and continue to predict, record discoveries, and search for patterns.



For a further explanation of star-finding, read "Stars and Primes" in David Thornburg's book, *Discovering Apple Logo*. For off-computer exploration of stars, see *Paper and Scissors Polygons*, by Linda Silvey and Loretta Taylor.

## Music

In the September LX, Judi Harris gives an interesting overview of Logo music. We sometimes provide the program for a familiar piece of music for younger children. (LogoWriter is used for this example, although other Logo versions have music capabilities, also.) The objective is simply to let them know that the computer can make sounds.

They are encouraged to turn to the FLIP side of the page to change various elements and to experiment with notes and phrases. The children can experimentally rearrange sections of the music to make a new tune.

Our music teacher revised the frequency chart in the Reference Manual to write this version of "O Christmas Tree." He even borrowed a piano tuner's frequency meter to determine the best-sounding tones and intervals for the program.

```
TO TREE
PLAY1
PLAY1'
PLAY2
PLAY1"
END
```

```
TO PLAY1
B 24 E 18 E 6 E 24 F# 24 G# 18 G# 6 G# 48
F# 18 G# 6 A 24 D# 24 F# 18 E 6 E 24
END
```

```
TO PLAY1'
B 24 E 18 E 6 E 24 F# 24 G# 18 G# 6 G# 48
F# 18 G# 6 A 24 D# 24 F# 18 E 6 E 42
END
```

```
TO PLAY2
B.HI 6 B.HI 18 G# 6 C#.HI 42 B.HI 6 B.HI 18 A 6 A 42
A 6 A 18 F# 6 B.HI 42 A 6 A 18 G# 6 G# 24
END
```

```
TO PLAY1"
B 24 E 18 E 6 E 24 F# 24 G# 18 G# 6 G# 48
F# 18 G# 6 A 24 D# 24 F# 18 E 6 E 48
END
```

```
TO C#.HI :DUR
TONE 1150 :DUR
END
```

```
TO F# :DUR
TONE 742 :DUR
END
```

```
TO B.HI :DUR
TONE 1013 :DUR
END
```

```
TO E :DUR
TONE 657 :DUR
END
```

```
TO A :DUR
TONE 888 :DUR
END
```

```
TO D# :DUR
TONE 617 :DUR
END
```

```
TO G# :DUR
TONE 838 :DUR
END
```

```
TO B :DUR
TONE 487 :DUR
END
```

If the Logo version you are using has a different command than TONE for music, just make the appropriate substitutions in the procedures above. Or, you could write a utility procedure, such as the one below for use with Apple Logo II.

```
TO TONE :FREQ :DUR
TOOT :FREQ :DUR
END
```

The children make a design appropriate for the musical selection. LogoWriter provides two ways to build a word design, using either the LABEL or word processing functions.

It is easier to use word processing for longer selections.

```

      *|*
    STAR
  *~*~*~*
HOLIDAYS
  *~*~*~*~*~*
DECORATIONS*
  *~*~*~*~*~*~*~*
  *RINGING BELLS**
  **~*~*~*~*~*~*~*~*~*~*
  **GLOWING CANDLES***
  ** SUMPTUOUS FEASTS **
  ** CAROLS FILL THE AIR **
  ** FRIENDS AND FAMILIES **
  *****
      NOEL
      ****
      NOEL

```

Happy Holidays!

*Elaine Blitman and Barbara Jamile are the K-2 and 3-4 supervisors at the Punahou School in Honolulu, HI. They have been using Logo with young children since 1982. Their CompuServe number is 76067,211.*

## Logo LinX

by Judi Harris

### Cooperative Means to Interdisciplinary Ends

Many educators find Logo to be a natural facilitator of group cooperative problem solving activities. I have also found it to be a natural facilitator of team teaching efforts. In this month's column, I plan to vary the focus of the article just a bit, from a Logo project idea to the cooperative development process that was its genesis.

"Genesis" is certainly an appropriate term here. At one of the two Solomon Schechter Day Schools (private religious schools) in Philadelphia, I had the good fortune to work with Sara Lynn Newberger, a knowledgeable and energetic fourth-grade Jewish studies teacher, and NancyLee Rodenberg Bergey, a most talented science coordinator.

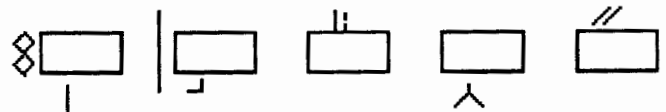
### Interdisciplinary Roots

Schechter was in the midst of a six-week interdisciplinary science exploration of "Shapes, Patterns, and Structures." The teachers of all subjects: general studies, Jewish studies, and specialties, were interweaving SSPS (Schechter Shapes, Patterns, and Structures) projects and activities into their curricula, facilitated and cheered on by NancyLee. The time was ripe for cooperative teaching, when NancyLee "just happened" to mention what the fourth grade was studying.

The joint project began almost accidentally. Sara Lynn was in the faculty room, describing the method that she was using to help her students understand the chanting melody notation system (trope signs) included in the Hebrew version of the book of Genesis. She was helping the children to focus on the patterns of the trope signs, so that they eventually would understand how the notations emphasized the meanings of the phrases to which they were attached.

### Logo Structures from Long Ago

Sara Lynn had drawn the symbols on flash cards, overhead transparencies, and worksheets, attached to rectangular boxes, that looked something like this:



The boxes represented whatever Hebrew words the trope signs were punctuating.

When I happened upon this faculty room conversation, Sara Lynn was describing how she would help the fourth-graders to "diagram" selected Hebrew verses from Genesis, then deduce their common trope sign patterns. Several of the teachers listening were interested in the organization of these patterns, and how they reflected the meaning of the Bible verses. As Sara Lynn described their hierarchical procedural structure, my "Logo light bulb" lit up.

### Logo ma non "trope"-po

What followed was one of the most rewarding experiences of my work with children and teachers. Sara Lynn had learned to use the Bank Street Writer for composing class progress reports and report card comments, and was happily planning to use the Hebrew version of the program ("Kosher Writer") but had not, as of yet, been bitten by the "Logo bug." Her students had been using Logo since second grade.

When I think back on it, Sara Lynn showed an immense amount of trust (an essential component in successful team teaching) by committing a good portion of her free time to discussions with this seemingly "possessed" computer specialist. I insisted that a Logo microworld could significantly help her students to accomplish the process and content goals that she had described in the faculty room. We were equally ignorant of each other's specialties, but equally interested in child-centered, holistic learning.



Our project planning discussions were as interesting as the children's interactions with the resultant microworld. We began by examining the structure of the trope notation system, drawing similarities to programming structures in Logo. This necessitated "reciprocal tutoring;" trope patterns for me, and Logo procedure patterns for Sara Lynn. Once content goals were established in both realms (I thought this the perfect way to encourage the students' writing of single-function procedures, then cooperatively arranging them in a hierarchical super-procedure structure), we began to discuss process goals in terms of what the students should be able to do with the microworld.

### The Microworld

Herein lies the full power of cooperative planning for teaching. By focussing upon the *actions* that the students would be able to select in the microworld (the "new primitives"), we two teachers were able to integrate the content's own structure (Torah trope patterns) into the organization of the microworld itself. The discourse was delightful. Each suggestion that either of us posed could be fleshed out in terms of how a specific student action would reflect powerful ideas in the study of trope patterns and/or Logo. Our respective ignorance of each other's specialties forced us both to be specific and lucid about our ideas. This probably helped to increase the quality of the microworld. Before we began this joint venture, we would have predicted just the opposite.

A cooperative project is a cooperative project. The students worked with the cantor (among whose synagogue responsibilities is that of chanting weekly Bible portions) to write Logo music procedures for each of the trope melodies. Sara Lynn and I constructed the microworld so that as student users type trope name abbreviations, they hear the melody and see the verse diagrammed on the screen. The microworld "remembers" the order of signs for each verse. When the diagram is finished, the computer prompts a chapter and verse number, then stores the diagram in procedure form on the disk. In this way, a database of verse diagrams is compiled by the students, which the user can then print out and reproduce in any combination as paper handouts and/or overhead transparencies. Sara Lynn's relative lack of computer experience helped me to make the program quite user friendly.

### Eight Steps to Success

I realize that few, if any, of you will have need for a Torah trope microworld in your classroom. The point here is not the product, but the process, and the possibilities of its generic application. (Sound familiar?) You may very well be the only staff member at your school that has any knowledge of Logo; yet, that certainly doesn't mean that you can't help other teachers to integrate Logo into their traditional (and nontraditional) content curricula. Similar microworlds can be cooperatively developed and shared in practically any subject area.

These guidelines may be helpful:

1. Decide upon the content goals of the microworld.
2. Decide upon the process goals of the microworld.
3. Determine how the content and processes selected are naturally organized.
4. Divide the organization into the smallest single-function user actions; these will be the microworld primitives.
5. Code the subprocedures so that they are as flexible, easy to understand, and easy to change as possible.
6. Encourage the potential users to help to program the microworld subprocedures.
7. Decide that the microworld will never be "finished;" revise it continuously according to user feedback.
8. Encourage other teacher and student users to change the microworld to suit their purposes and preferences.

Michael Friendly, the *Logo Exchange* North American Field Editor, is currently compiling a database of Logo microworlds. If you have developed any microworlds for use in teaching, or have ideas for some, consider sharing what you have on a wider scale. Please send microworld information to:

Michael Friendly  
York University  
Psychology Department  
Toronto, Ontario M3J 1P3  
Canada

### Timeless Wisdom

The theme of the national Conference on Alternatives in Jewish Education (CAJE) gathering at which I presented this project was, appropriately,

*I have learned much from my teachers, more from my friends, and from my students most of all.*

Originally written in Hebrew, this quote is probably more than one thousand years old. Its lesson still rings true.

May this holiday season inspire peaceful cooperative ventures for all peoples, and may we all share in the joy and respect that are implicit in such worthy endeavors.

---

*Judi Harris was an elementary school computer use facilitator, graduate education instructor, and computer consultant for a number of public and private schools in Pennsylvania. She is now a doctoral student in education at the University of Virginia. Her CompuServe electronic mail address is 75116,1207.*

---

# MathWorlds

edited by A. J. (Sandy) Dawson

## On Functions and Variables

Is it true that students learning a computer language will also learn some mathematics? The answer to that question, according to Frederick Klotz, a professor of mathematics at St. Patrick's College in Dublin, Ireland, seems to be yes, provided the appropriate computer language is used. Klotz writes:

"In some respects computer programming seems closely related to mathematics. Good programming requires the same combination of clear logic, attention to detail, and inventiveness that is characteristic of good mathematics. Yet many important concepts in programming -- loops, pointers, arrays, and floating point arithmetics, to name just a few -- are awkward to deal with in mathematical terms. Even the concept of a variable in computer science looks strange to someone accustomed to variables in mathematics..."

"However ... there is a style of programming, called functional programming, in which the connection with mathematics is direct and explicit. A functional language such as Logo, is ideally suited to the needs of the mathematics teacher." [1, pp.15-16]

Klotz contends that "...in a functional language, such as Logo or LISP, one interprets a program as an operation, formed from simpler operations by functional composition. Usually operations are functions in the strict mathematical sense. That is, they are rules which associate with each element of a certain domain a unique image... In functional programming the underlying model is the traditional function machine of mathematics. The emphasis is not on what is going on inside the computer, but on how to link up function machines the computer emulates to achieve a desired objective." [1, pp.16-17]

Some simple examples from Klotz's work are "...the absolute value function, defined in mathematical notation as follows:

$$\text{ABS } X = \begin{cases} X & X > 0 \\ -X & X \leq 0 \end{cases}$$

"The Logo version of this is ...

```
TO ABS :X
IF :X > 0 THEN OUTPUT :X
ELSE OUTPUT (- :X)
END
```

"We can use ABS as a building block to construct ... more complicated functions such as  $f(x) = \sin(|x|)$ :

```
TO F :X
OUTPUT SIN ABS :X
END
```

"(SIN is a built-in function in Logo). Is this the same as  $g(x) = |\sin x|$  for all values of  $x$ ? Define

```
TO G :X
OUTPUT ABS SIN :X
END
```

and the computer can help to provide the answer."

Klotz continues, "For many functions, inductive definitions are most convenient. Consider for instance  $XP$  for non-negative integers  $p$ :

$$XP = \begin{cases} 1 & p = 0 \\ X \cdot XP-1 & p > 0 \end{cases}$$

"In Logo we would write:

```
TO POWER :X :P
IF :P = 0 THEN OUTPUT 1
ELSE OUTPUT :X * POWER :X :P - 1
END
```

"POWER illustrates the programming technique of recursion. As this example shows, recursion serves as the computer equivalent of inductive definition." [1, pp.17-19]

The gap between computer programming and mathematics can be bridged, then, if functional programming and a functional language such as Logo is used. But how does one do that with secondary school children? Leron and Zazkis investigated this question. They worked with above average children, aged 14 and above. In their *Function and Variables Learning Unit* they focused on these topics:

- Functions (input and output, equal functions, and equivalent procedures, various forms, predicates).
- More functions (single-valuedness, domain of definition, identity function, constant function).
- Inverse functions
- Conditions of invertibility
- Composition of functions
- Equations, subprocedures, constraint systems and unknowns.
- Functions of several variables.
- The scope of variables. [2, p. 187]

They devised some Logo-based mathematical situations (illustrated on the following page) which would lead children to an awareness of the above concepts.

### What is a Function?

"We have here a few Logo procedures called PUZZLE1, PUZZLE2, etc., each with one input. You can play with them and find out what they do.

```
?PUZZLE1 2
5
?PUZZLE1 5
11
```

"Now, without the computer, can you guess what will PUZZLE1 4 be? Check your guess with the computer. Can you write a procedure (called ANSWER1) that will have 'the same behavior' as our PUZZLE1?" [2, p. 187]

These activities are reminiscent of Bob Davis' early work on the Madison Project when he invented "What's My Rule?" games. Leron and Zazkis make use of Davis' strategy of the finding "secrets" to deal with the topics of equations as functions and subprocedures as variables and unknowns. This is how they did it.

"Here is a function called FUN.

```
TO FUN :X
PRINT :X * 3 + 5
END
```

"Someone had executed FUN with an unknown value of :X and got 20 as a result. Find the unknown value of the input :X. This is a different, procedural way to present the equation  $3x + 5 = 20$ . One way to solve it is by inverting the function (i.e., the input  $\rightarrow$  output processing). Thus the sequence of operations :

$X \rightarrow :X * 3 \rightarrow :X * 3 + 5$

is inverted to :

$Y \rightarrow :Y - 5 \rightarrow (:Y - 5) / 3 \dots$

"We can thus write NUF, the inverse of FUN.

```
TO NUF :Y
PRINT (:Y - 5) / 3
END
```

"Since NUF 20 yields 5, we know that FUN 5 should yield 20. Thus 5 is the desired value of the unknown input."

### The FAN Game

"FAN is a procedure with a secret in it:

```
TO FAN
REPEAT 8 [ SECRET LT 45 ]
END
```

"Can you draw on a piece of paper the effect of FAN if SECRET is given by:

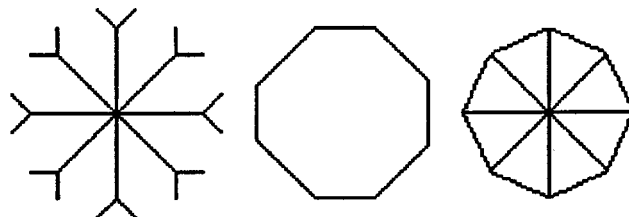
```
TO SECRET
FD 50 BK 50
END
```

"And what if we change SECRET to:

```
TO SECRET
FD 50
REPEAT 3 [ FD 10 RT 120 ]
BK 50
END
```

"In this game, FAN is a picture form whose result depends on the subprocedure SECRET. SECRET is therefore a variable subprocedure, and the above activity consists of evaluating the form under various substitutions for the variable.

"Here is the same game with a new twist. The following drawings are produced by the FAN procedure (with different 'values' of the 'variable' SECRET):



"Can you find the SECRET that produces each of these drawings? (Remember, you can only change SECRET, not FAN.)" [2, pp. 189-190]

The educational ideas which guided the work of Leron and Zazkis bear repeating here:

"-There was no use of ready-made software in this unit. The children's programs (as well as some of our own) were the object of study.

"-We used a wide range of functions both in variety and difficulty.

"-We tried to write as much as possible in 'soft,' everyday language preferring simple words and common sense explanations to technical language and jargon. In the same vein, we would sometimes adopt for several sessions the terminology suggested by the kids, before we introduce the standard terms.

"-The learning is all done through activities (some suggested by us, some of a free exploratory nature), followed by reflection on these activities through discussion, writing and some thinking pieces to take home.

"-We made a special effort to devise examples and activities that would take into account the children's point of view, in addition to the mathematical one. As much as possible, the activities should be interesting and the mathematics should emerge from them in a natural way.

"-Introduction of subtle points was delayed until they were actually needed, even at the cost of having the children temporarily develop buggy mental models." [1, pp. 191-192]

# **LOGO EXCHANGE**

## **The Magazine for LOGO Activities Worldwide**

Tom Lough, Editor  
Dennis Harper, International Editor

The *National Logo Exchange* and *International Exchange Newsletters* have been combined into a new magazine — **LOGO EXCHANGE**.

Now you have a dynamic magazine that is your personal window for professional Logo activities worldwide.

### Why Choose This Resource?

Regardless of whether you teach Logo or use Logo to teach, **LOGO EXCHANGE**:

1. Brings you exciting ideas from top Logo teachers throughout the world
2. Provides you with current information on Logo research, resources and methods
3. Is your personal window for professional Logo activities

### Regular LOGO EXCHANGE Features

Logo articles  
by classroom teachers  
for classroom teachers.

**Tips for Teachers**, a column by Steve Tipps, featuring ideas for using Logo in various content areas.

**Listful Thinking**, by Glen Bull and Paula Cochran, offers innovative applications of Logo list operations at a variety of levels.

**Turtle Tips**, by Donna Lanyi and Jane Toth, features Logo teaching ideas for the beginning teacher using turtle graphics.

**Q and A**, by Jim McCauley, provides thoughtful answers to Logo questions from **LOGO EXCHANGE** readers.

### What is in each issue of LX?

- Columns written by outstanding Logo field editors from each continent.
- Updates on international Logo research technology.
- Information on activities, meetings, and conferences of Logo organizations worldwide.
- Reports from our international correspondents.

### Additional Monthly Features

**LXual Challenges**, by Robs Muir, presents a Logo brainteaser in an intellectual and educational context.

**Teacher to Teacher**, sponsored by Logo information for Teachers, contains straightforward reviews about Logo books and other resources with comments from classroom teachers.

**LXionary** is a lectionary of suggested Logo readings, with commentary by Griff Wigley.

**Logo Disserts**, by Barbara Elias, contains information on recent Logo research, results, as reported in dissertations and other literature.

### Who should subscribe?

**LOGO EXCHANGE** is an international Logo resource for:

- National ministries of education,
- Universities and teacher training colleges,
- Libraries of higher education institutions,
- International educators,
- K - 12 teachers who desire a global perspective,
- Logo and computer education organizations,
- Research and development centers.

### What are the benefits of a subscription?

**Logo Exchange magazine will:**  
Bring you exciting ideas from top Logo practitioners around the world.

Provide current information on Logo research, resources, and methods.

Be your personal window for professional Logo activities worldwide.

### Subscription Information

**LOGO EXCHANGE** is published monthly except for June, July, and August. A subscription is \$29.95.

## ORDER FORM

☐ Please enter my subscription to **LOGO EXCHANGE** magazine @\$29.95.

☐ Check Enclosed

☐ Bill Me

☐ Credit card payment (circle one):

AMEX      MasterCard

VISA      Diners Club

Card No. \_\_\_\_\_

Exp. Date \_\_\_\_\_

Cardholder's Signature \_\_\_\_\_

Subscriptions outside the U.S.A., please add:

☐ \$ 6.00 per subscription for postage (\$15.00 air)



**Meckler Publishing Corporation**

11 Ferry Lane West  
Westport, CT 06880  
203-226-6967

Name \_\_\_\_\_

Organization \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

☐ Please bill my organization      Purchase Order No. \_\_\_\_\_

Authorized Signature \_\_\_\_\_

### SPECIAL NOTE

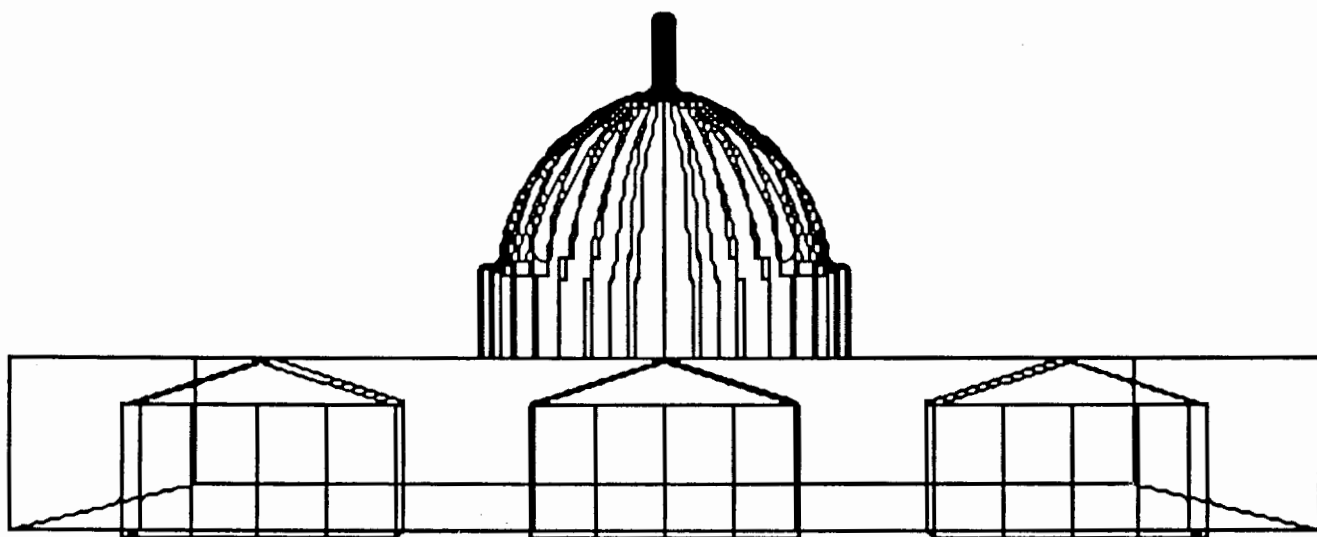
Meckler Publishing can now accept orders via electronic mail: ALANET (MECKLER.PUBS or ALA0959) or OnTyme (class: Meckler/JPE)  
ALANET subscribers may also order using an online order form. The form is accessible from the Library Supplier's Service Center Menu or from System level with the command REQUEST — Meckler.  
Those who wish to order via OnTyme please provide us with your complete name and address, your Meckler Publishing account number or credit card number and expiration date.  
Note author and short title for each Meckler Publication you are ordering online.

# EAST COAST LOGO CONFERENCE '87

ECLC  
'87



April 2 - April 4, 1987  
Stouffer Concourse Hotel  
Arlington, VA



A special meeting  
for everyone involved  
with Logo and computers in education.

**April 2, 3 & 4, 1987**  
**Stouffer Concourse Hotel**  
**Arlington, Virginia**

Hosted by the University of Virginia's  
Curry School of Education in association with  
the Meckler Publishing Corporation  
and *Logo Exchange* magazine

*Registration Information:*  
**11 Ferry Lane West, Westport, CT 06880**  
203-226-6967

This latter point is supported by Reinhold Wappler. He writes, "...children are not expected to learn suddenly all about variables and then happily use and debug them in their programs. We must 'allow for stages of incomplete knowledge', and fill these stages with appropriately graduated mental models and metaphors that are simple and useful to children, while at the same time consistent with future mathematical uses of variables." [3, p. 1]

Would Leron and Zazkis' approach to the topic of functions and variables via Logo prove more palatable to learners than the usual approach taken in the secondary school curriculum? Leron and Zazkis comment: "Is it Logo or maths? The kids were less enthusiastic to work on the 'functions and variables' unit than on Logo programming, but more enthusiastic than on mathematics. Conclusion: it is better to present such work to the kids as mathematics rather than Logo (if such a distinction must be made). Eventually, it seems that a mixed approach to functions would be best: Use Logo activities where and when they are interesting, relevant and beneficial." [2, p. 192]

### References

1. Frederick S. Klotz. "When is a Program Like a Function?" C. Hoyles, R. Noss, & R. Sutherland (Eds.). *Proceedings of the 2nd International Conference for Logo and Mathematics Education*, University of London, July 1986, pp. 14-24.
2. Uri Leron and Rina Zazkis. "Functions and Variables: A Case Study of Learning Mathematics through Logo." C. Hoyles, R. Noss, & R. Sutherland (Eds.). *Proceedings of the 2nd International Conference for Logo and Mathematics Education*, University of London, July 1986, pp. 186-192.
3. Reinhold Wappler. "Introducing Variables to Children: Language and Metaphors for an Intervention Strategy." Unpublished manuscript submitted to *Logo Exchange*, July 1986. The reference in Wappler's statement is from Uri Leron's article, "Logo Today: Vision and Reality," in the February 1985 issue of *The Computing Teacher*.

Sandy Dawson is a member of the Faculty of Education at Simon Fraser University in Vancouver, British Columbia. His CompuServe number is 76475,1315.

### Writing with Logo

Creative Word Processing in the Classroom announces *The Writing Notebook*, a bimonthly publication which focuses on writing at all levels. A regular LogoWriter column is one of the features listed in a recent press release.

Edited by Sharon Franklin (a former editor of *The Computing Teacher*) and Jon Madian, this periodical appears to present a well-balanced set of support materials for writing from elementary through college levels.

For more information, write to *The Writing Notebook*, PO Box 590727, San Francisco, CA 94159.

## SpecialTalk

by Paula Cochran and Glen Bull

### Touch Tools

This month we want to show how to make Logo respond to the touch of a switch -- whether it is an adapted joystick, a Koala Pad, or a switch designed especially for a handicapped child.

Using Logo with such peripheral devices can be a valuable option for some children, even if they are motorically able to use the regular keyboard. Switches are fun, and may provide a simpler and less distracting way of controlling Logo for non-handicapped as well as handicapped students.

Last month this column included directions for adapting a joystick for use with switches for motorically impaired users. We used the joystick project to show how a network of professionals can support each other through electronic communications. This month we will concentrate on the Logo mechanics of getting input from the keyboard or from a switch. Next month we will concentrate on related Logo applications which are being discussed and developed through the electronic network.

### Reading a Single Keypress

Often when you are writing a Logo program, you would like to include instructions for Logo to get some information from the user of the program. For example, Logo may need the answer to a question in order to make the rest of a procedure work correctly.

When the information needed is a single character (letter or number), the READCHAR command can be used to get input from the keyboard. (READCHAR and the other built-in commands discussed in this column are found in IBM Logo and Apple Logo II. There are equivalent commands in most other versions of Logo. Consult your program manual.)

The following procedure which uses the READCHAR command does not require the child to press a carriage return after answering Y or N for yes or no. Also note that the letter Y is quoted rather than enclosed in brackets in the line which checks the answer.

```
TO ASK.QUESTION.1
  PRINT [ DO YOU LIKE GRAPE JUICE MORE
  THAN ORANGE JUICE? ]
  PRINT [ PLEASE TYPE Y OR N. ]
  MAKE "ANSWER READCHAR
  TEST :ANSWER = "Y
  IFTRUE [ PRINT [ OKAY, YOU CAN HAVE
  GRAPE JUICE TODAY. ] ]
  IFFALSE [ PRINT [ FINE. YOU CAN HAVE
  ORANGE JUICE. ] ]
  END
```



The READCHAR command can be used to check whether the user has pressed a particular letter. Of course, the consequences of pressing the letter should be tailored to appropriate instructional objectives. The consequences might be moving the turtle a certain distance, as in many Instant Logo programs (see, for example, "Starting Logo with Young Children? Slow Down the Turtle!" by Dan Watt and Molly Watt in the September 1986 LX).

The example below could make good use of a SAY procedure. (Some computers have speech capabilities, and in some cases Logo can take advantage of this speech output. That topic needs another full column.) You can substitute a SAY procedure for PRINT if you have speech synthesis capabilities.

```
TO FIND.LETTER
PRINT [ CAN YOU FIND THE LETTER B? ]
MAKE "LETTER READCHAR
TEST :LETTER = "B
IFTRUE [ PRINT [ YES, YOU FOUND IT! ] ]
IFTRUE [ STOP ]
IFFALSE [ PRINT SENTENCE :LETTER [ IS NOT
THE ONE. ] ]
FIND.LETTER
END
```

A variation of this theme is a procedure which simply waits until the child presses any key. Such a procedure is very simple, but very useful, too. It does not test to see which key is pressed. It just waits for a key press, and then continues.

```
TO WAIT.FOR.KEYPRESS
MAKE "KEY READCHAR
END
```

The WAIT.FOR.KEYPRESS procedure is handy when you want to allow a reader to control the rate at which pages of text appear on the computer screen. We often name this procedure PAGE when it is used in Logo stories or projects. A screen of text can be printed, and then PAGE can be used to make the master procedure wait until a key is pressed before it continues.

### Cause and Effect

The WAIT.FOR.KEYPRESS procedure fits into a variety of Logo applications. We use synthesized speech with the example below, but you can substitute PRINT for SAY when you try it. This example also makes use of LOADPIC which is a built-in command to load pre-drawn pictures. Most versions of Logo have an equivalent for LOADPIC (the Terrapin equivalent is READPICK). The LOADPIC command loads a graphic picture into the computer monitor directly from a disk.

```
TO SHOW.PICTURE.1
SAY [ JOHNNY, TOUCH THE KEYBOARD TO
SEE GRAPES. ]
WAIT.FOR.KEYPRESS
CLEARSCREEN
LOADPIC "GRAPES
SAY [ THESE ARE GRAPES. ]
END
```

In this instance, the computer will say, "Johnny, touch the keyboard to see grapes." It will then wait for a keypress, and show a picture if a key is touched. The picture will fill the screen, and then the synthesizer will say, "These are grapes." If the computer did not have speech capabilities, the teacher could say the instructions instead.

### Sensitive to the Touch

Many computers allow up to two joysticks to be attached, and each joystick generally has two buttons. Thus, up to four buttons (or switches) can be sensed through the joystick input ports (gameports). Numbering in computers often starts with 0 instead of 1. Thus, the four buttons (two on each joystick) would be labeled Button 0, Button 1, Button 2, and Button 3.

Koala Pads (Koala Technologies Corp.) also attach to the computer through a gameport, and the two buttons on the top are equivalent to the buttons 0 and 1 on a joystick. Therefore, the procedures described below could be used with a joystick, an adaptive switch, or Koala Pad.

The built-in command BUTTONP is used to check whether a button has been pressed. Here is a procedure to show whether Button 0 has been pressed:

```
TO SHOW.BUTTON
PRINT BUTTONP 0
SHOW.BUTTON
END
```

This procedure prints true when Button 0 is being pressed, and false when the button is not pressed.

```
? SHOW.BUTTON
FALSE
FALSE
FALSE
TRUE <----- Button 0 is pressed at this point.
TRUE
TRUE
FALSE <----- Button 0 is released here.
FALSE
```

You will need to use an escape control key (such as Function-Break on an IBM PCjr or CTRL G in Terrapin Logo) to interrupt the SHOW.BUTTON procedure as it is written above.

The TRUE or FALSE reading from the BUTTONP command can be used to control a larger program. For example, a procedure equivalent to WAIT.FOR.KEYPRESS can be constructed using BUTTONP instead of READCHAR.

```
TO WAIT.FOR.BUTTONPRESS
TEST BUTTONP 0      (see if Button 0 is TRUE)
IFTRUE [ STOP ]
WAIT.FOR.BUTTONPRESS
END
```

The WAIT.FOR.BUTTONPRESS procedure works just like WAIT.FOR.KEYPRESS procedure. The only difference is that a button on a joystick or Koala Pad, or another external switch, is pressed instead of a key on the keyboard.

```
TO SHOW.PICTURE.2
PRINT [ JOHNNY, PRESS THE SWITCH TO SEE
GRAPES. ]
WAIT.FOR.BUTTONPRESS
CLEARSCREEN
LOADPIC "GRAPES
PRINT [ THESE ARE GRAPES. ]
END
```

Just as the WAIT.FOR.BUTTONPRESS procedure is equivalent to the WAIT.FOR.KEYPRESS procedure, the READ.BUTTON procedure below is similar to READCHAR.

If additional buttons are available, Buttons 2 and 3 can be added as additional lines in the READ.BUTTON procedure.

```
TO READ.BUTTON
IF BUTTONP 0 [ OUTPUT 0 ]
IF BUTTONP 1 [ OUTPUT 1 ]
OUTPUT READ.BUTTON
END
```

To try out this procedure, type the following:

```
? PRINT READ.BUTTON
```

Nothing will happen until you press the button or switch connected to the computer. Then either a 1 or a 0 will appear on the screen.

You can test for one of these numbers inside another procedure, to determine the consequences of pressing a certain switch or button. We have replaced READCHAR with READ.BUTTON:

```
TO ASK.QUESTION.2
PRINT [ DO YOU LIKE GRAPE JUICE MORE
THAN ORANGE JUICE? ]
PRINT [ PLEASE PRESS THE GREEN BUTTON
FOR YES. ]
PRINT [ PLEASE PRESS THE RED BUTTON FOR
NO. ]
```

```
MAKE "ANSWER READ.BUTTON
TEST :ANSWER = 0
IFTRUE [ PRINT [ OKAY, YOU CAN HAVE
GRAPE JUICE. ] ]
IFFALSE [ PRINT [ FINE, YOU CAN HAVE
ORANGE JUICE. ] ]
END
```

While the READ.BUTTON procedure is a user-defined procedure, it can be used in ways which are functionally equivalent to the built-in procedure called READCHAR (or RC in some versions of Logo). The teacher or clinician can label the buttons on a Koala Pad with red or green stickers, in the same way that the keys on the keyboard can be color coded. We hope you and your students enjoy making Logo work at the flick of a switch.

### SpecialTalk News

AMERICAN SPEECH - LANGUAGE - HEARING FOUNDATION COMPUTER CONFERENCE, FEBRUARY 18-22, 1987, HOUSTON, TEXAS. The fourth annual ASHF Computer Conference will be held at the Adam's Mark Hotel in Houston. Presentations and exhibits will emphasize current and future computer use in the treatment of communication disorders, and in related research. For more information, contact Nancy Minghetti, ASHF Program Director, 10801 Rockville Pike, MD 20852, (301) 897-5700.

---

*Paula Cochran is an assistant professor in the Communication Disorders Program of the University of Virginia's Curry School of Education. She is interested in Logo applications for language arts and special populations. Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. His CompuServe number is 72477,1637.*

---

## West Coast Logo Conference

The West Coast Logo and Telecommunications Conference will be held March 6 and 7, 1987, at the Los Angeles Airport Hilton Hotel.

Teachers, administrators, and others who use the Logo language or who have an interest in telecommunications are invited to plan now to attend this event. A major focus of the conference will be the integration of Logo into the curriculum.

For more information, write to:

West Coast Logo and  
Telecommunications Conference  
Pepperdine University  
3415 Sepulveda Boulevard  
Los Angeles, CA 90034

# Testudinal Testimony

by Douglas H. Clements

## Research on Logo and Affective Development

There is increased recognition of the important role that social and emotional factors play in children's learning. Last month we saw that Logo environments have the potential to encourage prosocial interaction. But what of students' affective, or emotional, side? Can Logo increase their self esteem? Their motivation to learn?

### The Goal

What do we want our students to develop in the emotional domain? Researchers demonstrate the importance of effectance motivation: the degree to which children desire to control or effect change in the environment. This subsumes several other constructs: self-concept; pleasure at intellectual discovery; independent, self-directed work; attitudes toward mathematics; and intrinsic motivation. Kamii (1985) proposes a related goal in a chapter entitled, "Autonomy: The Aim of Education for Piaget." Autonomy means being governed by oneself and making decisions for oneself. The mathematician Moise (1984) argues for the development of "psychic intelligence"—an inclination to use whatever cognitive intelligence one has for learning, for adaptive behavior, and for pleasure. To put it simply, we want students to want to use — and to believe in — their own minds.

### Self-Concept, Self-Confidence, and Commitment

According to their teachers, students working with Logo experience an increase in self-esteem and confidence ... if their teacher gives them greater autonomy over their learning and fosters social interaction (Fire Dog, 1985; Kull, 1986). These findings are echoed by Carmichael, Burnett, and Higginson (1985), who add that teachers reported such positive effects predominantly about students whose prior school experience was not a positive one. Researchers' observations of special needs children indicated that Logo provided them with prestige and respect from their peers, enhancing their self concepts (Michayluk, Saklofske, & Yackulic, 1984).

There is a good deal of qualitative evidence that some students who were previously not committed to their school work become intellectually involved and more self-confident working in the Logo environment, often transferring these new attitudes into the classroom (e.g., Badger, 1983; Findlayson, 1984).

Strand et al. (1986) showed that young students competently managed their computing environment and evidenced enthusiasm and confidence. Kindergarten children sustained their attention on Logo tasks for substantial time periods, even when they had the option to choose other activities in lieu of programming. Clements and Nastasi (1985) found that children

in a Logo environment exhibited a greater frequency of several behaviors indicative of effectance motivation, including engagement in self-directed explorations and showing pleasure at discovery. One researcher recorded the following statement of a boy persisting in the face of several setbacks on his Logo project: "He turns to me and very seriously says 'I say, never give up!'" (Carmichael et al., 1985, p. 286).

However, using a traditional quantitative instrument, Milojkovic (1983) reported that fifth graders engaged in computer programming scored significantly lower on one subscale of a measure of general perceived worth (as opposed to cognitive or social). We've seen that before: Observers see "good things" happening; tests show little or nothing.

Are enthusiastic teachers fooling themselves? Is it a "transfer problem"? Are the tests invalid or inappropriate? Should I extricate myself from this series of questions with the old standby: "More research is needed"?

While it is difficult to ignore the observational reports, neither can we disregard disappointing test results. We *do* need more research on these questions to learn how Logo can be used most effectively. For instance, in this case, could it be that the teacher-observers are building on their students' newborn pride and extending it to areas outside of Logo? There was little chance of this happening in the quantitative, 10 week, pull-out research project (Milojkovic, 1983). Other guidelines are provided by Carmichael et al. (1985), who identified key factors, under teacher control, that impacted on students' confidence and self-esteem. They included the degree to which teachers:

1. integrated Logo into their classroom;
2. balanced direction with exploration;
3. responded to the needs of their students;
4. encouraged individuals to share new discoveries and problems with the class and beyond; and
5. reduced competition or preferential treatment among individuals.

### Attitudes Toward Learning Mathematics

Students exposed to Logo, according to reports from teachers, are less bored in the classroom and exhibit more pleasure in their work (Fire Dog, 1985; Kull, 1986). A large scale evaluation project revealed that fourth graders trained in Logo had slightly less anxiety toward mathematics, and more confidence in learning mathematics, than had control students (Schwartz, Evans, & Caritj, 1984). Similarly, Clarke (1985) reported significant increases in positive attitudes toward learning mathematics in students (all girls) experienced with Logo. Blumenthal (1986) reported two studies in which Logo programming significantly increased low-achieving students' academic self-concepts.

However, other results contradict these positive findings. Horner and Maddux (1985) found no significant difference between Logo and control groups on a math attitudes scale. Perhaps inconsistent results should not be surprising, given that many students do not accept that Logo and mathematics are homologous enterprises. For example, Papert tells

of Kim, who hated math, but became engrossed in Logo programming. An unsuspecting visitor remarked to her that "computers make math fun." Kim looked up from her Logo work and spat angrily, "There ain't nothin' fun in math." In this case, her instructor had wisely not thought it advisable to discuss whether what she was doing was math. By the end of the year, she made the connection herself (Papert, 1980, p. 151).

Some do see the connection soon, however, and appear to like the new perspective. One junior said, "Logo is like learning math and having fun at the same time" (Carmichael et al., 1985, p. 288). The overall pattern is a positive, albeit slight and inconsistent, increase in student attitudes toward mathematics.

### Intrinsic Motivation

Milojkovic (1983) reported that fifth graders involved with Logo tended to take less responsibility for positive outcomes, but more for negative outcomes, a pattern characteristic of mastery-oriented students. They also placed a significantly greater value on a measure of independent judgment. This subscale produced the only significant difference on a measure of intrinsic motivation.

Similarly, Schwartz et al. (1984) found no overall difference on a scale of intrinsic motivation, however Logo students were more likely to use independent judgment. Clements and Nastasi (1985) found that children in a Logo environment were more likely to determine their own "rules" for interacting socially and for working.

Brown and Rood (1984) found positive effects on a measure of internalized locus of control. Likewise, Louie (1985) reported a significant shift towards an internal locus of control in children 12 years of age or less (the 16 hour computer experience, however, was confounded by combining Logo, word processing, and minimal arcade game activities). This shift was most evident in the areas of "outer control" and "social acceptance / group cooperation" (interestingly, the areas of "self-acceptance" and "authority responsiveness" showed a slight external shift, possibly due to students' feelings of being challenged by the computer and of needing adult assistance to meet the challenge).

Blumenthal (1986) reported that low-achieving children exposed to Logo programming did not score significantly higher on a measure of internality, although they did significantly outperform CAI and control groups on one subscale, attributing more of their successes on academic tasks to their own efforts. (Other subscales evidenced similar trends, but differences were not significant. This is perhaps not surprising given the length of the treatment—only 12 sessions.)

Horner and Maddux (1985) reported no significant increase on a locus of control instrument attributable to a 6 week experience with Logo. A significant difference was found, however, between the number of students observed as having internal attributions about Logo tasks. That is, work with Logo may have made students feel responsible for their success with the activities. If I have not confused you with

details, you may have discerned a consistent and interesting pattern: Although generalized effects were not consistently observed, students experiencing Logo did appear to judge situations for themselves and accept responsibility for their actions.

There are connections among intrinsic motivation, self-concept, and attitudes about mathematics that are potentially important to teachers. Harter (1978) has stated that children are initially intrinsically motivated to explore their environment. In so doing, they get pleasure from solving problems and learning. However, their beliefs about their own capabilities, or "perceived competence" depends on their receiving feedback from others. In other words, as they solve a problem, they need to hear from others how "well they are doing." Thus, Logo may motivate students to work, and to work autonomously.

However, teachers must help them see that they are tackling difficult tasks and gaining mastery over a complex endeavor. This is another reason teachers should not say, "Logo is easy," but rather, with Michael Temple, "Logo is hard, but it's worth it." Or, as a third grader put it, Logo is "very hard...but it had to be done. I liked doing it" (Carmichael et al., 1985, p. 90). Teachers should also provide praise which specifically informs students of the competencies they are developing.

Ultimately, intrinsic motivation, with proper teacher mediation, influences perceived competence. In turn, as students see and enjoy their own growth, perceived competence positively influences intrinsic motivation.

### Summary

Long-term studies are required to determine whether such gains are consolidated, but these findings do provide some evidence of Logo's power for enhancing effectance motivation. Next month we'll discuss "thinking together," or social cognition, in Logo environments. This in turn will lead us nicely back to the academics.

### References

- Badger, E. (1983). An evaluation of a Logo training program. Unpublished manuscript, The Commonwealth of Massachusetts, Department of Education, Quincy, MA.
- Blumenthal, W. (1986). The effects of computer instruction on low achieving children's academic self-beliefs and performance. Doctoral Dissertation, Nova University.
- Brown, S. W., & Rood, M. K. (1984, April). Training gifted students in Logo and BASIC: What is the difference? Paper presented at the meeting of the American Educational Research Association, New Orleans, LA.
- Carmichael, H. W., Burnett, J. D., Higginson, W. C., Moore, B. G., & Pollard, P. J. (1985). Computers, children and classrooms: A multisite evaluation of the creative use of microcomputers by elementary school children. Toronto, Ontario, Canada: Ministry of Education.

Clarke, V. A. (1985). The impact of computer on abilities and attitudes. Manuscript submitted for publication.

Clements, D. H., & Nastasi, B. K. (1985). Effects of computer environments on social-emotional development: Logo and computer-assisted instruction. *Computers in the Schools*, 2(2-3), 11-31.

Findlayson, H. M. (1984a). The transfer of mathematical problem solving skills from Logo experience. D.A.I. Research Paper No. 238. University of Edinburgh, Edinburgh, Scotland.

Fire Dog, P. (1985). Exciting effects of Logo in an urban public school system. *Educational Leadership*, 43, 45-47.

Harter, S. (1978). Effectance motivation reconsidered: Toward a developmental model. *Human Development*, 21, 34-64.

Horner, C. M., & Maddux, C. D. (1985). The effect of Logo on attributions toward success. *Computers in the Schools*, 2(2-3), 45-54.

Kamii, C. K. (1985). Young children reinvent arithmetic: Implications of Piaget's theory. New York: Teachers College Press.

Kull, J. A. (1986). Learning and Logo. In P. F. Campbell and G. G. Fein (Eds.), *Young children and microcomputers* (pp. 103-130). Englewood Cliffs, NJ: Prentice-Hall.

Louie, S. (1985, July). Locus of control among computer-using school children. Paper presented at the Logo 85 conference, MIT, Cambridge, MA.

Michayluk, J. O., Saklofske, D. H., & Yackulic, R. A. (1984, June). Logo. Paper presented to the CAP Convention, Ottawa, Ontario.

Milojkovic, J. D. (1983). Children learning computer programming: Cognitive and motivational consequences. Unpublished doctoral dissertation, Stanford University.

Moise, E. E. (1984). Mathematics, computation, and psychic intelligence. In V. P. Hansen & M. J. Zweng, (Eds.). *Computers in mathematics education (1984 Yearbook)*. Reston, VA: National Council of Teachers of Mathematics.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books.

Schwartz, T. A., Evans, H., & Caritj, W. H. (1984, April). Looking into a large-scale Logo program. Paper presented at the meeting of the American Educational Research Association, New Orleans, LA.

Strand, E., Gilstad, B., McCollum, P., and Genishi, C. (1986, April). A descriptive study comparing preschool and kindergarten Logo interaction. Paper presented at the annual meeting of the American Educational Research Association, San Francisco, CA.

---

*Doug Clements is an associate professor in the College of Education at Kent State University in Kent, OH. He teaches Logo courses and conducts research on the use of Logo to develop cognitive, mathematical, and social-emotional competencies. His CompuServe number is 76136,2027.*

---

## Logo Disserts: Dissertations Dealing with Logo

by Barbara Elias

Is it possible to develop an efficient way to analyze what children encounter in a Logo learning environment? In an attempt to understand the processes inherent in such an environment, Sharnee Chait used the technique of protocol analysis to study children's programming. (Protocols are detailed records of an activity.) This study resulted in a classification system which can be used to delineate the various phases in programming, to represent programming structures, and to identify programming errors and coping strategies.

Five seventh-grade students (three girls and two boys) participated in the study. They were selected because of their average to less-than-average grades in mathematics on their most recent report cards. None had had previous computer experience. They studied Logo, individually with the experimenter for about an hour twice a week for ten weeks.

Sessions were conducted during the regular school day, and were organized around Papert's teaching model. Students usually started projects with a drawing and were encouraged to break down projects so that subprocedures could be used.

The Logo system used in this study could generate a complete record of each child's work in a dribble file and replay the sessions for review. These dribble files and detailed observations provided data for the analysis of each child's work.

To develop the classification system, a bottom-up approach to data analysis was utilized. In such an approach, the system fits the data rather than the reverse. First, each set of data was divided into episodes. Episodes began when a child started something new or began a new session. Within each episode, a classification system was used to describe each child's activities. Six distinct kinds of activities were observed. They included:

1. Constructing and drawing
2. Generating and testing hypotheses for debugging
3. Procedure writing
4. Revising by editing
5. Adding instructions by editing
6. Activities which appeared to lack goal direction

This system was also used to describe the relationship between episodes, the child's evaluation of his or her work, bug types, and the debugging actions used.

A variety of bugs and debugging actions were exhibited by these children. Angle bugs were most often corrected by incremental adjustment. Direction bugs were remedied by reversing commands. Size and stop bugs were corrected by editing. Clearing the screen and starting over were processes used for position bugs.

Procedural nets which are used to represent programming structures diagram the major projects of each child and

add a considerable amount of clarity to the analyses. They show graphically the hierarchical structure of the programs, the complexity of programs, and the individual differences in programming.

This methodology is suggested as a first step approach to developing evaluation procedures for computer-based learning environments such as Logo. Although the methodology has certain shortcomings, it also presents techniques which are efficient in terms of time, have some categories which are well defined, and can be used to raise questions and isolate areas of Logo learning for further study. Teachers and older students may find adaptations of procedural nets beneficial for analyzing programming tasks.

>>>Chait, Shamee (1978). *An Analysis of Children's Problem Solving in a Graphics Oriented Computer Programming Environment*. (Unpublished thesis, Master of Arts, McGill University, Montreal, 1978).

(Ed. note: Each month, Barbara Elias highlights a dissertation or thesis dealing with Logo. She is accepting copies of recent research results for the LX Dissertation and Thesis Repository. Reports may be mailed to Barbara Elias, 4223 Hickory Road, Ettrick, VA 23803.)

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

### East Coast Logo Conference Scheduled for April 87

The 1987 East Coast Logo Conference (ECLC 87) is scheduled for April 2 - 4, 1987, at the Stouffer Concourse Hotel in Arlington, VA. Hosted by the University of Virginia, in association with Meckler Publishing Corporation, ECLC 87 will focus on Logo classroom applications and the extensions of Logo into areas of related technology.

Featured speakers will include Seymour Papert, Dan and Molly Watt, Paul Goldenberg, and Robert Tinker. Presentations by classroom teachers will concentrate on integrating Logo into the curriculum. A special session of student presentations will be included.

A series of presentations on Logo extensions will include connecting Logo to other hardware (such as videodisk players, speech synthesizers, and the like), educational methods and techniques which have a philosophy similar to that of Logo, an overview of Logolike software, and topics relating to the future of Logo.

To be placed on the ECLC 87 mailing list, send your name and address to:

East Coast Logo Conference  
Tom Lough  
Curry School of Education  
University of Virginia  
Charlottesville, VA 22903

## LIFT's

# Teacher to Teacher

by Anne Cairns Federlein

The book, *Computer Environments for Children*, by Cynthia Solomon should be purchased by every educator, administrator and parent! It is published by The MIT Press (28 Carlton Street, Cambridge, MA 02139) for \$22.95.

The 173-page text addresses new issues in education that have arisen since the introduction of the computer in the classroom. How the computer is used today depends on the theoretical orientation of the teacher. For example, those teachers schooled in drill and practice and rote learning will tend to teach using these methods. Their classes use computers as textbooks.

However, educators who understand child development use the discovery method. This teaching method allows the child to be in control of the learning with the teacher as a facilitator in the classroom. Here, the computer is integrated as a learning tool to be experienced, explored, and manipulated as the need occurs.

Two questions pervade the educational system today:

1. How do children acquire new knowledge?
2. What do we want them to learn and why?

Depending on the orientation and training of the teacher, these questions have different answers.

The book is divided into chapters on Computers in Education, Drill and Practice and Rote Learning, Socratic Interactions and Discovery Learning, Eclecticism and Heuristic Learning, Constructivism and Piagetian Learning, Trends in Practice, and Computer Educators. It concludes with an extensive bibliography with excellent references.

Solomon begins the book with a challenge by stating,

*For educators, a nagging question is, How do I learn to use computers today in a way that will not be obsolete in five years? The answer requires a vision of the future. But how can we develop a vision of the future? Journalists and business people might try to build their vision of the future by projecting statistics on how many computers schools are using today. Alas, school administrators often follow their example. To project to the future we need to understand more deeply what happened in the past.*

The author studies the relationship between people and computers. The machine can control the child when used as a textbook and or as a tool the child uses to extend learning. In the first example, a dumb machine controls a smart child while in the second, the smart child uses a dumb machine to get smarter!

Examples of those using the textbook concept are Patrick Suppes at Stanford University, author of sequenced drill



and practice exercises and Robert Davis, of the University of Illinois, who designed a Socratic set of exercises.

Using the experience method, Tom Dwyer, of the University of Pittsburgh, developed programming activities using BASIC. Seymour Papert and the Logo Group at Massachusetts Institute of Technology are involved in the study and implementation of teaching mathematics and technology using constructivism. A number of Logo examples are included in the text.

Solomon discusses teachers using the discovery method of teaching.

*She knows (and can only know this through experience) when to intervene and when to let the learner struggle. She believes that the key goal for any learner is to improve his image of himself as a learner, as an active intellectual agent.*

This statement sums up the pragmatic part of teaching using this style. However, the use of the pronouns (the teacher being identified as female, and the learner as male) could be offensive to male teachers and to females capable of using computers to learn...

The conclusion of the book offers challenges to allow active learning shared by both teachers and students. Solomon asks what activities in the learning centers will allow children to learn in an improved way? What new learning environments will emerge in the coming years?

These questions are important ones for anyone using Logo. Teachers and students developing new learning environments will answer these questions.

In the meanwhile, time passes...

---

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

---

## SeeLogo Announced

K-12 MicroMedia has announced the release of SeeLogo for the Apple II family of computers. Written in Pascal, SeeLogo's turtle graphics offers a variety of useful commands, including the ability to mix text and graphics.

SeeLogo is available to schools for \$59.95. This price includes a language disk and backup, plus a certificate entitling the school to purchase additional SeeLogo copies at a price of \$8.00 each.

A version for the home is available for \$39.95.

For additional information, call (201) 825-8888, or write to:

K-12 MicroMedia  
6 Arrow Road  
Ramsey, NJ 07446

## LXionary A Lectionary of Selected Logo Readings

*with Commentary and Opinion  
by Bill Craig*

>>>"On Computers, Personal Styles, and Being Human: A Conversation with Sherry Turkle." by Lewis A. Rhodes, *Educational Leadership*, March, 1986.

I am often guilty of paying so much attention to the gritty details of my daily responsibilities that I fail to step back and look at the "big picture." Sherry Turkle's comments forced me to examine some of the assumptions that guide my thinking about Logo, instructional computing, and education in general. The interview covers teacher training, the computer revolution, equal access, the importance of programming instruction, learning styles, teaching styles, and children's concept of being human. While Logo is only a small part of her comments, the issues she discusses are same issues that we struggle with in the Logo literature and in our Logo debates. I cannot summarize every important point that she makes but I will share the ones that seemed most interesting.

1. The statement that "kids don't like computers," is meaningless. It is the learning environments that are created with computers that students (and teachers) are reacting to, not the machine.

2. Computers should be given to teachers before we can expect the machines to be used with students. Along with the hardware, Turkle would provide four pieces of software: a data base program, a spreadsheet program, Logo, and a word processor. The feelings that teachers experience when they struggle with a new piece of software and the success they have in using software productively will allow them to identify with their students' struggles and successes.

3. Turkle's plan for a classroom revolution has two prongs. The first prong provides computers for teachers as described above. With students, computers would be introduced through word processing. Students would learn about computer operations, would become to see the computer as a productivity tool and may even become better writers.

4. Unlike television, computers increase one's attention span. Computers *take* our attention because they require that we put so much into them.

5. "I know empirically that not creating environments where there is permission for different styles to be legitimate leads to discrimination against women."

6. Turkle has an interest in the metaphysical implications of the computer for children, the effect of the computer on child's concept on what is life and what is not. She contends that working with the computer forces children to think about what is special about being human. Many children contend that while computers "think," humans "feel."

I cannot do justice in one or two sentences to most of the points that Turkle makes in the interview. I used to believe that while discussions of these issues were interesting, they had little importance for me as a classroom teacher. I was willing to leave the "philosophizing" to the college crowd. As I read the article, I gained a greater appreciation of the relevance of these debates. But more importantly, Sherry Turkle reaffirmed my belief in our potential to create exciting and meaningful environments for our students. Now that I have stepped back and looked at that big picture, the challenge is to translate this potential into what happens in classrooms everyday. Back to the gritty details.

*Bill Craig is the Computer Education Program Specialist for the Chesterfield County Schools, Chesterfield, VA.*

## Logo Notes

Software de Base announces LogoSB, a Spanish Logo for the IBM PC and compatibles. LogoSB includes features such as 3-d turtle, multiple turtles, and the ability to switch from Spanish to English. The language packet includes a language disk (copy protected), a reference manual in Spanish, and a summary with instructions in English. The price is \$120.00, postpaid. Software de Base, Antonio Cavero 39, Madrid Spain 28043.

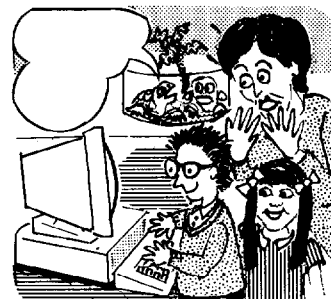
AI Logo for the IBM PC is now available from Ascent Incorporated. This version is command compatible with Apple Logo, including the use of the <Control> keys. The package includes a language disk (not copy protected) and a reference manual for \$88.00. For additional information, write to Ascent Incorporated, 190 Sobrante Way, Suite 201, Sunnyvale, CA 94086, or call (800) 367-5867. In California, call (408) 720-9200.

## The Adventures of Jacques and Elsie

*drawings by  
Linda Sherman*

In our October cartoon, Elsie was making a comment to Jacques regarding the reaction of the teacher to the Logo work of two students. Our favorite caption was submitted by Ceil Humphreys of Tampa, Florida.

**"Why didn't they teach us how to do that in school?"**



In this month's cartoon, Elsie is dreaming about interacting with her computer counterpart. What is she saying in her dream?



Please send your caption suggestions no later than December 22, 1986 to: Jacques and Elsie, Logo Exchange, PO Box 5341, Charlottesville, VA 22905. Include your name and address so that we can give proper credit should your caption be chosen.

All captions become the property of Meckler Publishing. None can be returned.

*Linda Sherman is a freelance author and artist living in Shipman, VA, with her husband and two-year-old son.*

Q

and

A

by  
Jim

McCauley

Q. I have run into a Logo problem that I don't understand. It concerns the primitive ERASE. In the 1.0 and 1.3 versions of Terrapin (MIT) Logo, ERASE will only operate on a single procedure. I wanted to write a procedure called ERASE.PROCEDURES which would erase as many previously defined procedures as desired. The procedure would accept a list of these procedures by way of the variable :PROCEDURE.LIST. I tried the following set of commands:

```
TO ERASE.PROCEDURES :PROCEDURE.LIST
IF :PROCEDURE.LIST = [] THEN STOP
ERASE FIRST :PROCEDURE.LIST
ERASE.PROCEDURES BF :PROCEDURE.LIST
END
```

Much to my surprise, I kept getting error messages stating that ERASE didn't like FIRST as an input. No matter what I tried, I couldn't get anything to work. Finally, I found a procedure that worked, but it used the primitive LIST, as follows:

```
TO ERASE.PROCEDURES :PROCEDURE.LIST
IF :PROCEDURE.LIST = [] THEN STOP
RUN LIST "ERASE FIRST :PROCEDURE.LIST
ERASE.PROCEDURES BF :PROCEDURE.LIST
END
```

Can you explain this for me? Are there other primitives that must use the RUN primitive? (I assume, that the object acted upon must be in the form of a Logo list.) I found that DEFINE requires this. Are there others?

A. You got error messages stating that ERASE didn't like FIRST as an input because of an interesting difference between MIT Logo and the LCSi implementations. Most primitives in MIT Logo require that any *words* that they receive as arguments must be *quoted*. For example:

```
?PRINT "FRED
FRED
```

If the word is not quoted, Logo will try to *evaluate* the word; that is, it will try to run the word as a procedure. For example, if you define a procedure named FRED which outputs the list [ HELLO THERE! ], here's what happens:

```
TO FRED
OUTPUT [ HELLO THERE! ]
END
```

```
?PRINT FRED
HELLO THERE!
```

But if there is no procedure named FRED, you'll get an error message.

In MIT Logo, there are some words that are excluded from this rule. EDIT is one of them, and TO, PO, and ERASE are the others. It is possible to give ERASE an unquoted word as input, and ERASE will carry out its proper action without first trying to evaluate

```
?ERASE FRITZ
(erases the MIT procedure named FRITZ)
```

In LCSi Logo, however, words used as inputs *must* be quoted. (There is only one exception -- the defining word TO.)

```
?ERASE "FRITZ
(erases the LCSi procedure named FRITZ)
```

At first glance, it seems like LCSi Logo is nitpicking. Why not save a keystroke and do it the MIT Logo way? The reason for the stringency is to avoid problems like the one you have run into. When you pass the FIRST element in the list :PROCEDURE.LIST to ERASE in MIT Logo, ERASE gets confused. It is looking for a plain old unquoted word, but when it "sees" FIRST as its input, it assumes that you want to erase the primitive FIRST, and erasing primitives is not allowed. LCSi Logo avoids the problem by being a bit more strict about quoting, and the result is that, in those dialects, ERASE will accept output from FIRST or any other operation.

You were able to overcome the problem by putting ERASE and the output of FIRST :PROCEDURE.LIST together with LIST, then applying RUN to the new structure. This is a good solution. As you can see, there really is a point to LCSi Logo's pickiness about quotes!

I hope that these remarks have helped you to understand the situation better.

Each month, I will answer a selected question related to Logo. If you have a question, send it along with a stamped self-addressed envelope to:

Jim McCauley  
Logo Exchange Q and A  
876 East 12th Street, #4  
Eugene, OR 97401

---

*Jim McCauley is a graduate student at the University of Oregon, studying with David Moursund, and has written Logo articles for many national publications. His CompuServe number is 70014,1136.*

---

# IntLXual Challenges

by Robs Muir

## Fuzzy Definitions

Riddle: What is a camel? Answer: A horse designed by a committee.

If language and intelligence can come in two different "flavors", I'd like to propose that, similarly, organizations of things or ideas can also be categorized as either artificial or natural. Such organizations often produce results or outcomes pleasing to the eye, the ear, or the mind. Whether pleasing or not, these resulting designs may also be classified as either "artificial designs" or "natural designs".

Objects in the real world usually generate designs of the natural sort. The growth spiral of a conch shell, the path a projectile falling under the influence of gravity, the shape a turtle creates when told to REPEAT 360 [ FD 1 RT 1 ], each of these designs is determined by forces outside the object's influence. The designs created by iron filings surrounding a magnet are dictated by electromagnetic forces present on an atomic scale; the magnet is not said to *actively create* the design.

On the other hand, many of the organizations of things or ideas that have been built by humankind produce designs which can be categorized as "artificial." The design of computer language is decidedly artificial; the design decisions here are often arbitrary. Human beings actively create the design. And yes, Virginia, Logo is an artificial design!

When I was first learning Logo [Terrapin (version 1.0) and, later, Apple Logo II], I was fascinated by the seemingly arbitrary and capricious nature of Logo's design. Was anyone else first drawn to those parts of the language labeled as "Miscellaneous and Advanced Commands" by Hal Abelson? Even better, how about the "Advanced or Rarely Used Primitives" (from Chapter 13 of the Apple Logo Reference Manual)? If LABEL / GO was such a travesty, why was it included in the language? Why do we need REPARSE? And what about COPYDEF? Surely there were some secrets the designers weren't letting us in on! As proof, there were even some "Magic Words"! Hmmm...deep secrets here! What *do* you use XZZZY for?! (See Apple Logo Reference Manual, p. 142)

Words are power. Just as big numbers denote more power for children than do smaller numbers ( RT 876 FD 12706 ), "advanced or rarely used primitives" promised great power for me. Perhaps here was the "Holy Slow Road" to Logo mysticism; perhaps these words were the key to conjuring Logo magic. Better yet, perhaps this knowledge would turn mystic arts into science -- prestidigitation to programming.

In a sense, I was right; my instincts were right. I was also not alone. Many people, as they learned Logo, also gravitated to the last chapters in the Logo manuals for a taste of arcane knowledge and to gain further insights into how Logo works. A precious few books even dare to delve into these regions within this non-natural language.

Fred Achsberger, a fellow Logo learner, helped build a classic Logo program that I continue to use in some of my more advanced workshops. It uses a "rarely used" primitive

present in LCSi Logos to great advantage. Additionally, this program helps turn even high priests of Logo into novitiates again. It gives a teacher the opportunity to become a learner; a role we seldom get to play as an adult. It is also a great trick to play on your jaded 10-year-old Logo students.

Called MAZE, the program is simple. A maze is drawn on the screen, the turtle is placed at the start of the maze, and the user is invited to write a procedure called SOLVE that will pilot the turtle out of the maze. The magic here is to *redefine* the way FD, BK, LT, and RT work.

FD 50 does different things depending on the current heading of the turtle; RT is not always the obverse of LT. What this means is the old, familiar turtle commands don't work the way you expect. This is a situation a first grader may encounter as they first begin to use Logo; "It doesn't work right!" With MAZE, a near-fatal case of cognitive dissonance engulfs the normal adult Logo expert; "It doesn't work right!" Now the learning takes place, as we remap new behaviors on familiar commands. It is also a great place to do "science" - build hypotheses to explain the as-yet unexplained.

Can you build a program similar to MAZE? How cleverly can you hide the text of each newly redefined FD, BK, RT, and LT command? Do you know where to begin?

The tool to use is either COPYDEF or DEFINE. The syntax is as follows:

**COPYDEF "NEWNAME "FORWARD** will create a procedure called NEWNAME and give it the same definition as FORWARD.

**DEFINE "ONWARD [ [ NUM ] [ FD :NUM ] ]** will build a new procedure called ONWARD having one input (:NUM) which then behaves like the primitive FD.

In Apple Logo, neither COPYDEF nor DEFINE can be used to "redefine" a primitive, i.e., DEFINE "RT [ [ num ] [ RIGHT ( :num \* 3 ) ] ], *UNLESS* you first change a global system variable (or "flag") called "REDEFP. Yes, Virginia, this is a "magic" word! Before you redefine RT, type

**?MAKE "REDEFP "TRUE**

You can now redefine or "clobber" the primitive set to your heart's content, until you MAKE "REDEFP "FALSE.

Brian Silverman and Michael Tempel of LCSi have demonstrated similar Logo ideas within the framework of the "Fuzzy Logo" microworlds.

If you'd like a copy of Fred's MAZE program, please send your version (working or not) and we will trade listings. Enclose a self-addressed, stamped envelope and send your solution to this month's challenge to:

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

*Robs Muir is a physics and computer science teacher in Claremont, CA, and an instructor at the Claremont Graduate School. His CompuServe number is 70357,3403, and his Bitnet address is MUIR@CLARGRAD.*



# International Logo News



Dennis Harper  
Logo Exchange International Editor

## Global Comments

by  
Dennis Harper  
Institute of Education  
469 Bukit Timah Road  
Singapore 1025, Republic of Singapore

This month's international selections in the *Logo Exchange* find descriptions of Logo activities in Iceland, India, Argentina, and Brazil. Recently, Iceland has been very much in the news and the *Logo Exchange* has a permanent correspondent in Reykjavik (eat your hearts out Time, NBC, etc.). For a country of only 250,000 people, Iceland's Logo activity is as hot as its geysers.

Few countries could be as opposite as Iceland and India (population around 600,000,000 or one Iclander for 2400 Indians). The Asian column tells about 750 Indian schools who are engaged in a large computer literacy pilot study; the programming language of choice for this initial venture into secondary school computer education -- Logo. The third international article is equally exciting with the description of a local Logo center catering to 250 people weekly and a huge South American Logo Congress. I am pleased to see such positive and encouraging reports from the international field.

## Computers and Society

I would like to continue with my discussions of the problems encountered when trying to introduce computer education and Logo into developing nations. In the past two issues of the *Logo Exchange*, problems dealing with teacher training and educational hierarchies were explored. The remainder of this column will discuss briefly the problems in the society that influence the introduction of computers into education.

Obvious problems are a high illiteracy rate, lack of communication networks (few and untrustworthy phone systems), and the false impression that computers are not needed in industry, agriculture, or business. Other natural conditions and cultural distractions include such things as frequent holidays, power outages, monsoons, coups, heat, humidity, dust, insects, and illness. In addition, family, tribal, and religious commitments compete for students' time to a greater extent in developing countries (in general) than in western nations.

Graft, kickbacks, and crime also work against extensive computer education efforts. There are examples of computer distributors breaking up hardware and software into parts and selling them separately. Documentation (photocopied) is sold separately and often key pages are missing; these pages are sold for an additional fee. The cost of a standard IBM PC has risen as high as US \$14,000 in some countries. An additional problem of importing educational computing materials is the hefty duty imposed. Also, bribes to customs inspectors are sometimes just a part of "business as usual."

The high cost of equipment means that most computers are owned by the nation's rich and by the country's elite schools. Fortunately, some of these problems are being rectified, but the road to computer literacy is very long indeed for many developing nations. The dream of a computer for each child sometimes seems quite unlikely.

This issue marks the end of the first year of the *International Logo Exchange*. The first four issues were in the separate ILX newsletter and the last four have been incorporated into the present format. I would like to thank all my continental editors for their long hours and hard work. I would like to encourage you, the *Logo Exchange* readers, to send to the continental editors information about Logo activities which should be reported to our international readership.

FD 100!

## Europe

by

Richard Noss

London University Logo Group

Institute of Education

Department of Math, Statistics, &amp; Computing

20 Bedford Way

London, England WC1 HOAL

## Logo News from Iceland

reported by Anna Kristjansdottir

It all started in MIT at Logo 85. Several people began discussing the idea of a tour for Logo enthusiasts to the Netherlands and Iceland. Sure, I was willing to help organize and so I did. It was quite interesting, not least because we have been using Logo in our teacher education and inservice training for a few years. Besides that, substantial work has been put into making translations of different versions of Logo into Icelandic. Why was that done? -- you might ask. Surely not because the children couldn't understand English. No, understanding English does not seem to be the trouble. The "problem" is that a nation with only a quarter of a million inhabitants really cares about its delicate old language and tries to avoid the massive invasion of foreign words.

So there we were, having given a basic foundation in Logo to approximately 10-15% of Icelandic teachers and having Icelandic translations almost ready. Everything was set for a Logo course in August during the visit of the American group. We were really looking forward to it. This seemed to be a splendid opportunity to really start some Logo work in the schools on a sound basis.

But this didn't seem to be the right year for an American visit to Europe - peculiarly enough not even to Iceland. [I can easily understand this given the "imminent danger" of a terrorist attack on Reykjavik - R. N.]. So in April, I was awakened at five o'clock in the morning by a call from the USA telling me that, in spite of all efforts, there were simply not enough people signing up for the tour, so it would have to be cancelled. That was it.

What could I do? Knowing that it wouldn't be particularly interesting for our teachers to settle for pure Icelandic leadership of the course, I decided to contact Richard Noss. There were two reasons why. Richard was, after all, the contact person in Europe for the *International Logo Exchange* newsletter. Also, the BBC Acorn computer is quite common in Icelandic schools, so it was not a bad idea to have somebody from England. And Richard did his job to our great satisfaction. But he couldn't come...

That's how it happened that Professor Celia Hoyles came to Iceland and directed a week-long course on Logo. With all her experience through the Logo Maths project at the University of London Institute of Education, she gave us very valuable insight into the development of concepts and understanding among children learning Logo.

The teachers participating in the course had many different backgrounds, some teaching in the primary range, others in lower secondary or even upper secondary, some having done no Logo work with children. Others were experienced mathematics teachers, specialists in foreign languages, and some had been in the computer field for a long time. It is quite a challenge to cope with a course of that kind.

All the teachers in the group started off with hands-on experience constructing tessellations, and an amazing variety of patterns emerged. As part of this activity, the teachers learned about procedures, modularity, and took time out to play "little people." A whole day was set aside for recursion and the teachers used this in a wide range of projects of their own choosing. Throughout the course, children's work from the Logo Maths Project was discussed and educational issues were woven into the work as appropriate. Celia gave some specific lectures on particular programming aspects, and also more general ideas on the use of the computers in education, policy changes in the UK, and current research issues such as teacher inservice, gender, and developments in Logo itself.

The last day was spent sharing the teachers' own projects, and talking about what would happen with Logo in the future in Iceland. The course was a real success according to the participants, and it was a pleasure to work with Celia. On behalf of the educators in Iceland who participated in the course or worked there with Celia, we would all like to thank her warmly for the effort she made in creating an inspiring course.

## Visitors to the USA

In October, a group of four Icelandic educators visited the USA as participants in a program sponsored by the Eisenhower Exchange Fellowships, Inc. The program theme was the use of computers in education, in particular, innovative projects and their implementation, with an emphasis on students between the ages of 12 and 19. One of the topics for investigation was the application of Logo for various purposes and with various age groups.

Mr. Jon Torfi Jonasson, of the University of Iceland, and a member of the Eisenhower group, spent part of his visit in Charlottesville, Virginia, meeting with Glen Bull, Paula Cochran, Tom Lough, and several others involved with the Logo work there.

For more information about Logo in Iceland, write to:

Anna Kristjansdottir  
Vesturgata 34  
101 Reykjavik, Iceland



Asia  
by  
Hillel Weintraub  
Doshisha International High School  
Tatara, Tanabe-cho, Tsuzuki-gun  
Kyoto-fu, Japan 610-03

In our last LX Asia report, we learned something about Logo's presence in China, the world's most populous nation. It seems fitting to turn our attention now to India, the country with the second largest population in the world.

At the end of August, I visited Tokyo to attend the IFIP Regional Conference on Microcomputers in Secondary Education. I was fortunate to meet Professor S. N. Maheshwari of the Indian Institute of Technology in Delhi. This month's report is based on information from his conference presentation. We got together over coffee to talk specifically about Logo in India. He showed me the manual for a Logo package, called MATHLAND, which was developed at his institute by one of his students, Sandeep Kumar, as part of his graduating thesis.

In the first of this two-part series, I would like to present an overview of computer education at the undergraduate level in India. In the next report from Asia (to appear in the February 1987 LX), the focus will be more specifically on Logo. I think that it is instructive and important for those of us living in other countries to know about the development of India's computer education program, since it started from scratch just two years ago under an extremely controlled and centralized approach, a situation very unlike the way computer use has been evolving in North America.

### Computers in India: an Overview

In July 1984, the Indian government launched a program entitled "Computer Literacy and Studies in Schools," known as CLASS. This pilot program was to last for two years and then be the basis for a wider program. The CLASS objectives were:

1. Give the students "a broad understanding of computers and their use."
2. Provide hands-on experience with some aspect of computers.
3. Give the students a familiarity with a wide range of computer applications and an understanding of the computer's potential as a tool for processing information and controlling various devices
4. Demystify computers and familiarize students to the point that they would be able to express themselves creatively in developing applications "relevant to their immediate environment."

In order to meet these goals, the following major modules were developed:

1. Programming with Logo (turtle graphics only);
2. Using a database package;
3. Working with modelling and analysis using a spreadsheet program;
4. Word processing;
5. Working with a computer graphics package.

Three teachers from each of the 250 schools selected for the CLASS project were given three weeks of training, followed by continued support from a regional training center.

Each school was provided with 2 BBC Model B computers with color monitor, floppy drive and 1 dot matrix printer. They were also given software to support the computer literacy goals mentioned earlier (Logo turtle graphics, word processing, data base and spreadsheet management) and also open-ended simulations for computer based learning in subjects like chemistry, physics, math, biology, and geography.

Teachers were encouraged to take the computers into their regular classes and experiment with the various software. This was an attempt to involve many teachers beyond the three especially trained and many students beyond those chosen for the CLASS project.

As a result of CLASS project, a home grown educational software industry has developed. Indigenous packages for turtle graphics (which we will be looking at in depth next time), a database system, a spreadsheet program, and a word processor have replaced the packages used initially. In addition, various CBL packages which directly support the school curriculum have been developed.

One of the problems noted by the evaluation team is the lack of time available for computer projects within school hours. Traditional curriculum and traditional teaching styles have a strong hold in India, as in most other countries. In addition, school administrators have been reluctant to replace library time or play time with CLASS project hours, saying, quite correctly I believe, that these activities are necessary for the students.

The original goal of the project called for a weekly lecture / demonstration combined with much emphasis on hands-on practice, leading to some rethinking about traditional educational methods of passing information directly from teacher to student with little chance for exploration. However, due to the pressure from within the school system and social system, this traditional approach has ended up being reinforced rather than broken down, because the weekly lecture / demonstration session has been held regularly, while the students have been getting "much less" than the recommended two hours of practice per week.

Because of this, plus the reinforcement of traditionally held biases (technology being male dominated and science / math oriented), the project planners, as well as the teachers, are convinced that the CLASS activities should be integrated broadly within the curriculum.

This has come about as teachers see the potential of turtle graphics in geometry study and systematic problem solving. The use of various information retrieval systems are also felt to be important in the upper grade subjects because of their potential to change the role of the teacher from information giver to helper of students learning how to use information. Teachers are also becoming aware of the potential for computers to simulate complex processes in ways not possible in most school labs, allowing students to become analyzers as well as observers.

It is now felt that this program will be a "seed and catalyst" bringing about important changes in how education is seen and implemented in schools in India. There is no doubt in anyone's mind that computers in secondary schools are here to stay and will, over time, "fundamentally change the nature of the education process."

For further information, contact:

Professor S. N. Maheshwari  
Department of Computer Science and Engineering  
Indian Institute of Technology, Delhi  
New Delhi 110 016, India

### Latin America

by

*Eduardo Cavallo and Patricia Dowling*  
*Instituto Bayard*  
*Salguero 2969*  
*1425 Buenos Aires, Argentina*

### Opening of a Public Logo Workshop in Buenos Aires

The first public Logo workshop in Argentina opened on August 19th in Buenos Aires. This workshop is another expression of a computing culture which is growing in Argentina and it is open to the public in general without restrictions as to age or intellectual capacity.

The idea of "taking the computer out to the street" is an old wish of Dr. Antonio Battro, currently president of the Asociacion Amigos de Logo of Buenos Aires. It has become possible thanks to the efforts of the Buenos Aires City Council, which lent a beautiful room in the cloisters of the Buenos Aires Cultural Center, the Asociacion Amigos de Logo which is taking charge of the equipment and the coordination of computing activities, and the Buenos Aires Cultural Center Foundation, which is acting as sponsor.

On the opening day we talked to Mr. Hector Thompson who has been director of the workshop since its creation. We learned about interesting details of its organization and activities. From December 1985 when the project started, up to May 1986 when it was opened to the public, a group of instructors was trained. Actually, that group is made up of 22 people, from 10 to 65 years old. They are in charge of those who attended the workshop since the May opening. Presently,

250 people a week work at the Logo room, paying a small fee towards the upkeep of equipment. Several commercial companies collaborated to obtain the machines which have increased in number from 3 to 13.

The opening day activities were attended by the Secretary of Culture of the Buenos Aires City Council, who made the opening speech. Then Dr. Battro told about what had been done and future plans. Horacio Reggini gave a talk concerning "forms in space." After the words by Hector Thompson, we could visit the workshop and share in an opening toast.

We hope this kind of activity will spread swiftly, to the benefit of a more democratic culture.

### III International Logo Congress:

#### Informatics in Education

Porto Alegre, Brazil, November 6, 7, 8, 1986

An important Latin American Logo event took place in Brazil this past month. The event attracted teachers, students, technicians and people interested in Logo philosophy and language.

The congress was sponsored by the Cognitive Studies Laboratory of the Psychology Department of the Federal University of Rio Grande do Sul - Porto Alegre, the EDUCOM project of the Ministry of Education, the Secretary of Education and Culture of Novo Hamburgo and the Editorial Group SINOS. UNESCO also backed the Congress.

The aims of the congress were:

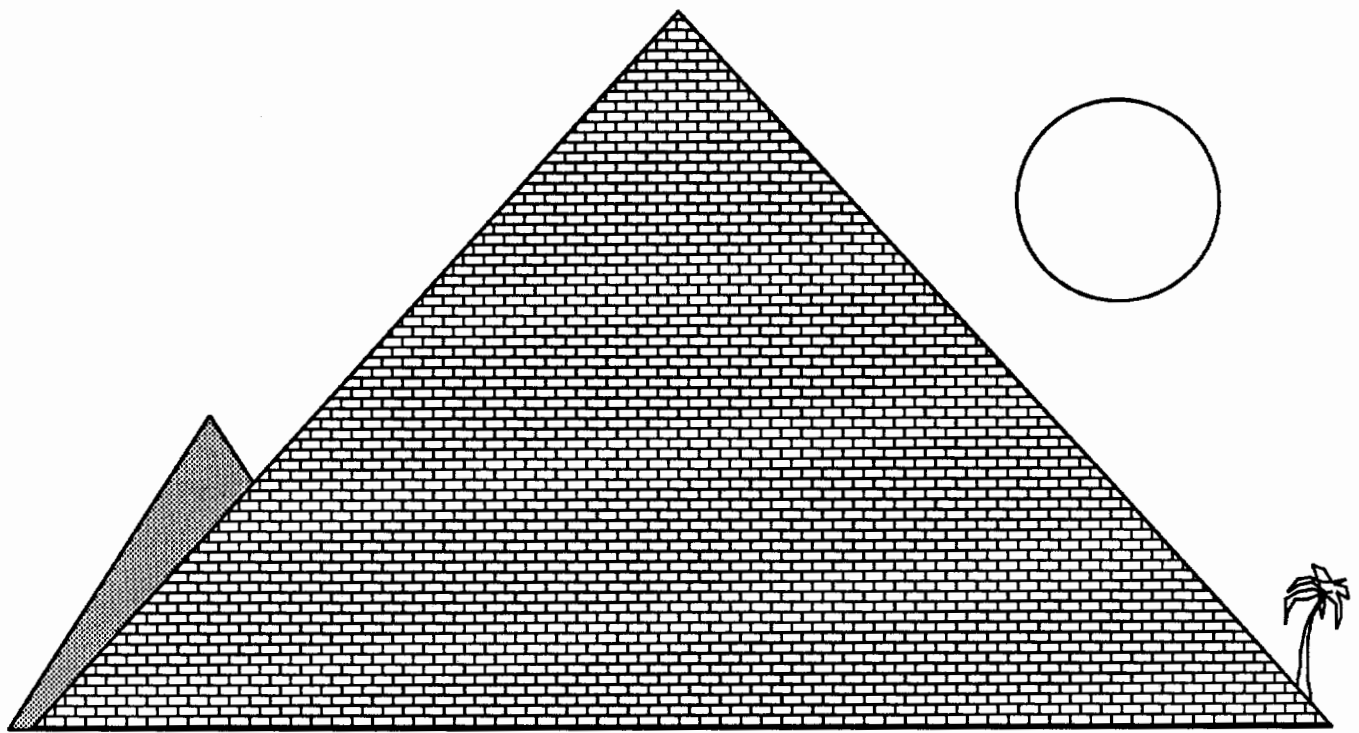
- to know, discuss, analyze, and debate about the use of Logo
- to study new perspectives and alternatives in education
- to spread new Logo teaching methods
- to discuss the results of application of the Logo culture
- to exchange research results and plan further research

Honorary President of the Congress was Dr. Seymour Papert and the general coordination was in the hands of Ernesto Sarlet, Secretary of Education and Culture of Novo Hamburgo, Lea Da Cruz Fagundes, from the EDUCOM project, and Paulo Sergio Guzman from the Agora project.

The themes of the Congress included Logo in Primary, secondary, special, artistic, and teacher education, as well as Logo in the cognitive sciences. Different implementations of Logo were also highlighted.

Activities included international and national symposiums, papers, descriptions of experiences, technological exhibitions, and workshops.

The congress was a very positive experience and was a new step towards the extension of a Logo culture in Latin America. In our next column, we will have the opportunity to share more of our experiences in Brazil.



“If something’s not broken, don’t try to fix it . . . support it!”

- Ancient Turtylonian

Terrapin™ Logo, from the original Logo people, is the Logo of choice in education. Terrapin has developed the **LogoWorks™** series of curriculum support materials including *Logo Works: Lessons in Logo* and *The Logo Project Book: Exploring Words and Lists*. Terrapin is also developing a host of Logo tools and applications for its new **Logoware™** product line including *The Logo Data Toolkit*—many more are on the way. So you can be sure that Terrapin Logo is the best possible Logo to use as a foundation for *your* choice of Logo activities—to suit *your* classroom needs. Plus, with Terrapin’s 10-Paks, 20-Paks, networked version and district pricing, Terrapin is the best value for your money. Terrapin. The support you’ve been looking for!

***Terrapin Logo***  
***The Original Logo People***

Terrapin, Inc., 222 Third Street, Cambridge, MA, 02142 (617) 492-8816

# **\*\* Teacher's Pet \*\***



*"...the best robot I've ever seen..."*

—Seymour Papert, M.I.T. LOGO '85 Conference, Cambridge, Massachusetts.

*"We have found the Valiant Turtle's presence has increased the interest level of students and teachers in Logo, facilitated the understanding of Logo commands, and changed personal interaction during Logo work, providing students more time to share and discuss their observations."*

—June 1986 review in *The Computing Teacher* by Jim Ellickson-Brown, Mountain View Elementary School and Doris Mimnaugh, 4th grade teacher, Wilson Elementary School, Corvallis, Oregon.

Meet Valiant Turtle, state of the art in Logo programming! Controlled from your Apple II+, IIe, IIc, IBM PC, PCjr or Commodore 64 computer by an infrared beam, the Valiant Turtle requires no cords or wires to move. Create a design on the screen and watch Valiant draw it on paper! Watch as Valiant executes your commands in three dimensions. Plan some fancy footwork and choreograph a dance in Logo!

Valiant Turtle is the ideal learning tool for students learning Logo as well as a great introduction to the world of robotics. Valiant is simple to use and easy for even the very youngest Logo learner to understand. Complicated and sophisticated ideas are presented in simple, graphic form. Students build artificial intelligence concepts and learn to think about space and spatial relationships.

Valiant Turtle operates with most popular versions of the Logo language. Watch through Valiant's transparent plexiglass dome as the Valiant Turtle draws with extreme accuracy with its built-in pen. Valiant comes completely assembled with easy-to-read instructions and control software. Because it's so easy to set up and use, the Valiant generates much enthusiasm in the classroom! Young imaginations fly whenever Valiant Turtle enters the room!

The Valiant Turtle is available from Harvard Associates, Inc., 260 Beacon Street, Somerville, Massachusetts 02143. Harvard Associates provides full technical support for the Valiant. For more information, or to order your Valiant Turtle, please call (617) 492-0660.