

---

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

---



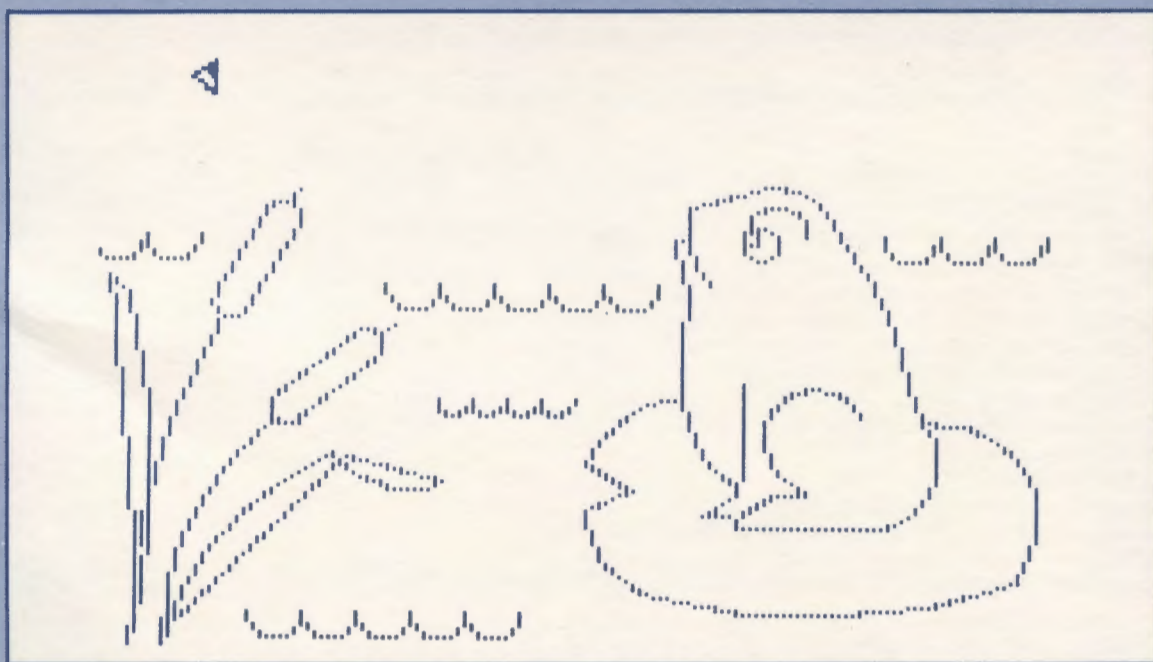
# LOGO EXCHANGE

---

SEPTEMBER 1988

VOLUME 7 NUMBER 1

---



---

International Council for Computers in Education



Publications

# NECC '89 Connections

## 9th National Educational Computing Conference *Invites you to attend*

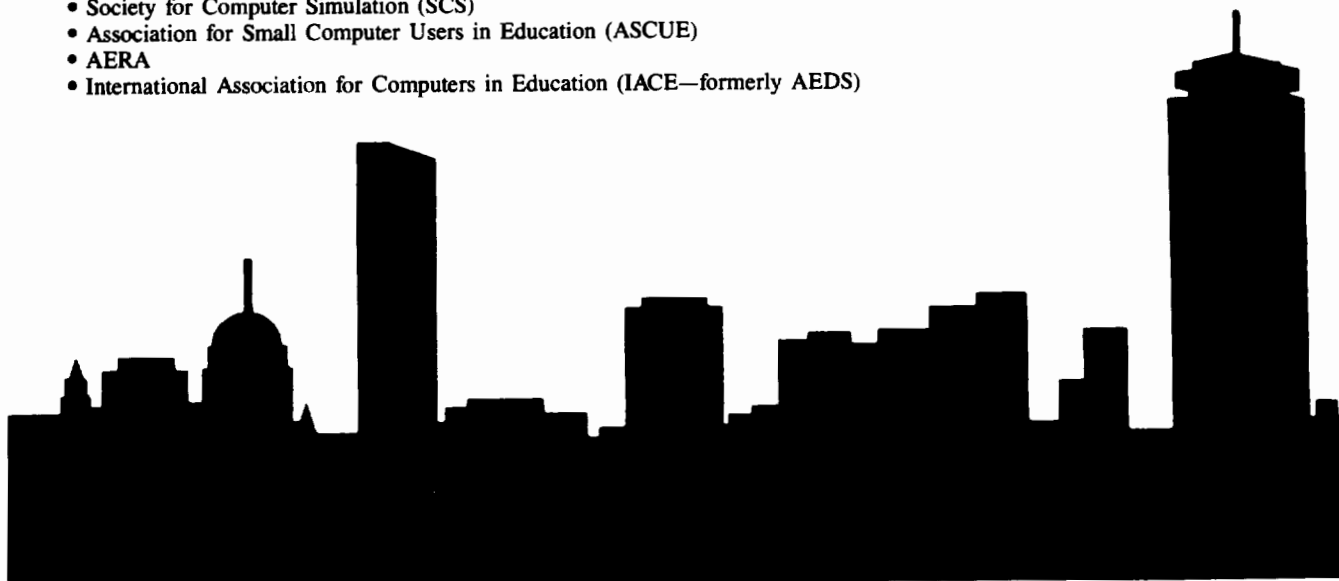
Papers, Panels, Projects, Exhibits, Full-Day Pre- Conference Workshops

For information contact:  
Susan Friel and Nancy Roberts  
Lesley College  
29 Everett Street  
Cambridge, MA 02138-2790

For information on Exhibits contact:  
Paul Katz  
NECC '89 Exhibits  
University of Oregon  
Continuation Center  
1553 Moss Street  
Eugene, OR 97403-9905

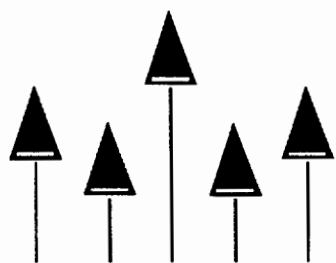
NECC '89 is sponsored by the International Council for Computers in Education (ICCE) in cooperation with member organizations of the NECC Steering Committee:

- Association for Computers and the Humanities (ACH)
- Association for Computing Machinery (ACM) Special Interest Groups on:
  - Computer Science Education (SIGCSE)
  - Computer Uses in Education (SIGCUE)
  - University and College Computing Services (SIGUCCS)
- Educational Computing at Minority Institutions (ECMI)
- IEEE Computer Society
- EDUCOM/EDUNET
- Society for Computer Simulation (SCS)
- Association for Small Computer Users in Education (ASCUE)
- AERA
- International Association for Computers in Education (IACE—formerly AEDS)



**Boston, Massachusetts**

**June 20-22, 1989**



# LOGO EXCHANGE

VOLUME 7 NUMBER 1

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

September 1988

## Founding Editor

Tom Lough

## Editor

Sharon Burrowes Yoder

## International Editor

Dennis Harper

## Field Editors

Anne McDougall  
Richard Noss  
Harry Pinxteren  
Fatimata Seye Sylla  
Jose Armando Valente  
Hillel Weintraub

## Contributing Editors

Eadie Adamson  
Gina Bull  
Glen Bull  
Doug Clements  
Sandy Dawson  
Judi Harris  
Gary Stager  
Leslie Thyberg  
Dan Watt

## International Council for Computers in Education

Anita Best, Managing Editor  
Kathleen Geygan, Advertising Director  
Dave Moursund, CEO  
Keith Wetzell, SIG Coordinator

## SIGLogo Board of Directors

Peter Rawitsch, President  
Gary Stager, Vice-President  
Ted Norton, Communications  
Frank Matthews, Treasurer

## Publisher

International Council for  
Computers in Education

*Logo Exchange* (ISSN 0888-6970) is the journal of the International Council for Computers in Education Special Interest Group for Logo-using Educators (SIGLogo), published monthly except for June, July, and August by ICCE, University of Oregon, 1787 Agate Street, Eugene, OR 97403-9905, USA.

POSTMASTER: Send address changes to *Logo Exchange*, UofO, 1787 Agate St., Eugene, OR 97403. Second-class postage paid at Eugene OR. USPS #000-554.

## CONTENTS

|   |    |
|---|----|
| From the Editor -- <i>Take Some Time</i><br>Sharon Burrowes Yoder   | 2  |
| Monthly Musings -- <i>Forward and Back: A Special Trip</i><br>Tom Lough   | 3  |
| Little Kids and Logo -- <i>A Bushel and a Peck</i><br>Leslie Thyberg  | 4  |
| Logo Ideas -- (Almost) <i>All You Need to Know About<br/>Subdirectories in LogoWriter V2.0</i><br>Eadie Adamson | 7  |
| Stager's Stuff -- <i>Another Bunch of Election Stuff</i><br>Gary Stager   | 11 |
| LogoLinux -- <i>Secular Conversions</i><br>Judi Harris  | 15 |
| MathWorlds<br>Sandy Dawson  | 18 |
| Assessing Logo Learning in Classrooms --<br><i>What Are Your Logo Students Learning?</i><br>Dan Watt            | 20 |
| Logo Connections -- <i>The Journey from Lego to Logo</i><br>Glen Bull and Gina Bull                             | 23 |
| Variables and Problem Solving<br>Dave Moursund and Sharon Burrowes Yoder  | 26 |
| Search and Research -- <i>Early Experiences with Mathematics</i><br>Douglas H. Clements                         | 30 |
| Global News<br>Dennis Harper  | 33 |
| The Value of a Group Project for Teachers of Logo<br>Charles R. McNemey   | 35 |

## SIGLogo Membership (includes *The Logo Exchange*)

|   | U.S.  | NON-U.S. |
|---|-------|----------|
| ICCE Member   | 24.95 | 29.95    |
| Non-ICCE Member   | 29.95 | 34.95    |
| ICCE Membership<br>(includes <i>The Computing Teacher</i> ) | 28.50 | 31.50    |

Send membership dues to ICCE. Add \$2.50 for processing if payment does not accompany your dues. VISA and Mastercard accepted.

© All papers and programs are copyrighted by ICCE unless otherwise specified. Permission for republication of programs or papers must first be gained from ICCE c/o Margaret McDonald Rasmussen.

Opinions expressed in this publication are those of the authors and do not necessarily reflect or represent the official policy of ICCE.

## From the Editor

### Take Some Time

Early this summer I had the opportunity to see the United States, from ocean to ocean. In mid-May, I packed my belongings and my cat, Moonwatcher, into a 14 foot U-haul truck, put my car on a dolly and began my journey to my new home in Oregon. The trip began in downtown Manhattan where we packed the truck midst honking taxis, rushing New Yorkers, and busy freight elevators in the 50 story building in which I lived. Once packed, I drove out of the City, connected with Interstate 80, and headed west. Soon I found myself in familiar territory in Pennsylvania and then Ohio, where I had taught for over 10 years. After a brief stop in Ohio to see friends and relatives, Moonwatcher and I continued west, following I-80 all the way to Utah where we picked up I-84 and finally connected with I-5 in Portland on the last leg of our journey to Eugene.

It goes without saying that driving provides an entirely different perspective from flying. The sheer hours of travel made me acutely aware of the size of this country. The plains of Nebraska and the mountains of Utah provide a striking contrast to the streets of New York City. I saw rolling hills, mountains, and endless plains. I saw farm land and ranch land. I saw cities and villages. Every hour brought a change in scenery as the miles and the hours rolled by.

I had to make frequent stops for fuel, particularly in hilly or mountainous areas. Still in my New York hurry-up mode, I initially found these interruptions frustrating, until I realized that each stop connected me to the people who lived in the part of the country where I had stopped. The "pit stops" varied from highly efficient gas station-assembly lines, to little one-pump general stores. The attendants varied from young to old, their accents and dress reflecting their environs. As I began to slow down, I found myself contemplating how alike all these people are, and yet how different. How strange the New Yorker's bustling would seem to the quiet young woman who helped me in Utah. How foreign the relaxed, friendly atmosphere in Nebraska might be to the busy attendants in Cleveland.

With lots of time to think, I found myself contemplating my career changes from the public schools to private industry, and now to the university. I thought a lot about education in general. I thought about the schools that the children I saw attended. How different they must be. I thought about their teachers and the tremendous responsibility they have to convey what I was experiencing of this country to their charges. Increasingly the educational system about which we often complain seemed like a miracle. How does it work at all in such a variety of settings?

And what about technology? Were computers in those schools? What were they used for? How many of the classrooms used Logo...and what did they do with it? Were the Logo drawings of the rural student in Iowa dramatically different from the urban student in Pennsylvania? As I pondered those differences, I found myself re-realizing that Logo is a kind of universal language. Young people from different cultural settings may draw different pictures, but they all understand how to move the turtle.

I know that my words can't adequately convey the sense of wonder that I felt as I arrived in Oregon. In a world condensed by jet travel, we forget the wondrous details of our own city or country. We are often in too much of a hurry to appreciate what is right about where we live, be it a village in Africa, a small town in South America, or a city in Canada. Make it your goal this year to take some time from your own rat race. Look at the best in yourself, your school, your students, and your community. Think about how you can effect positive change? Take some time to be in awe of the world, and share your wonder with others. (Perhaps your contemplative moments can produce the core of an optimistic classroom Logo project. I'd love to hear from you if you give it a try!)

Since my arrival in Eugene, I've spent my summer falling in love with Oregon and learning about all facets of publishing *LX*. If you've never put a newsletter or magazine together, it's hard to convey how much is involved in getting 32 pages to you every month! The ICCE staff has been a wonderful help, teaching me what I needed to know and helping me get caught up so that we can get Volume 7 out on time — indeed a miracle for you who have patiently endured late issues of *LX* all year.

As we begin this new year, I certainly hope that you will find *LX* even more helpful and enjoyable than in the past. I particularly want to welcome our two new columnists, Dan Watt and Leslie Thyberg. Dan is going to address the issue of assessing Logo learning, while Leslie will provide ideas for those of your working with primary aged children. I'm sure you'll delight in these two additions to *LX*. Like last year, I would love to hear from you. Send an article — we can work on refining it together. Send some student work for the cover. Write me a letter and tell me what you like and don't like about *LX*. But most important, take some time for reflection. My days of travel and thought were a wonderful opportunity for redirection that I will value for many years to come.

*Sharon Burrowes Yoder, 1787 Agate Street, Eugene,  
Oregon 97403, CIS 73007,1645 BitNet ICCE@Oregon*



## Monthly Musings

### Forward and Back: A Special Trip

by Tom Lough

Welcome back! I can hardly believe this is the beginning of the seventh year of publication! And we now have more subscribers than ever. But, as you will see in the next several issues, we are just getting started.

One of the most significant reasons for such rosy health is the energy of the ICCE's Special Interest Group for Logo (SIGLogo). At the SIGLogo meeting held during the 1988 National Educational Computing Conference in Dallas, TX, there was no doubt that the SIG was alive, well, and growing. The bylaws were discussed and adopted. The Logo Exchange magazine was designated as the official SIG publication, I am proud to report. Several exciting projects were planned. One such project involved exploring ways to use telecommunications for sharing Logo information among SIGLogo members. If you have any ideas or suggestions for this, please send them to me at the address below. And, naturally, if you are not yet a member of SIGLogo, I'd like to suggest that you join immediately. As a member of SIGLogo, you will receive your very own LX copy each month!

\*\*\*\*\*

A friend once told me, "Sometimes you have to take a step back before you can go forward." At the time the remark was offered, it was in the context of a complicated personal situation. Upon later reflection, I noticed a bit of Logo in it also. In fact, that bit of Logo has led me to do some additional thinking which I would like to share with you.

On the face of it, the remark seems to suggest a procedure similar to the following.

```
TO ADJUST :AMOUNT
BACK :AMOUNT
FORWARD :AMOUNT
END
```

After some thought, I decided that a related procedure would be much more useful, and seemed to contain more grist for the mental mill.

```
TO FB :AMOUNT
FORWARD :AMOUNT
BACK :AMOUNT
END
```

The FB procedure does essentially the same thing as

ADJUST, but the commands are in a different order. It is state transparent; that is, the turtle has the same position and the same heading after the FB procedure is run as it had before. It is an interesting utility procedure, helping you to draw flower patterns

```
TO FLOWER
REPEAT 36 [FB 30 RIGHT 10]
END
```

and produce lines with tick marks on them.

```
TO TICKLINE :LENGTH :DIVISIONS
REPEAT :DIVISIONS
[TICK FORWARD :LENGTH / :DIVISIONS]
TICK
END

TO TICK
LEFT 90
REPEAT 2 [FB 10 RIGHT 180]
RIGHT 90
END
```

An outstanding teacher in the Los Angeles area named Parnelli Sharp developed a catchy little song for preschool and kindergarten children to sing. One of the lines goes, "The turtle moves forward and back." The FB procedure in action!

Even though the turtle is going nowhere in the FB procedure, I discovered that this can be used to my advantage in some situations. For example, sometimes I need to draw a line from the turtle's current position to a spot I can see on the screen. I do not know the coordinates of the spot nor do I know how far it is from the turtle. I want a tool which will allow me to "play" with various lines until I find the one I want. With a small change, FB gives me that tool.

```
TO FB :AMOUNT
FORWARD :AMOUNT
WAIT 10
PENERASE
BACK :AMOUNT
PENDOWN
END
```

Now I am able to explore lines of different lengths before drawing the one I select. Also, I can now adjust the heading of the turtle between FB's, and get the line oriented exactly as I want it. I have used the above FB procedure numerous times with success. Moreover, I find that many Logo beginners have enjoyed using it, too, since they often want to erase a line they have drawn. Give it a try for yourself.

## Monthly Musings --continued

It was much later in the game that I paid more attention to the name of the procedure. I had selected FB because of the FORWARD and BACK commands. However, the letters FB also reminded me of the word "FeedBack." By using the FB procedure as a probe of sorts, I was getting feedback on the various lines, enabling me to adjust the heading and line length until I had exactly what I wanted. Then I realized something.

Many times in my life, I have felt as if I were going nowhere. Everything I tried to do was met with frustration. The phrase, "Butting your head against a brick wall," took on a special meaning. Progress seemed impossible. But, somehow, I decided to make adjustments between attempts. And, sooner or later, things started to happen, just like using the FB procedure. Sometimes it was a breakthrough with a student who was not motivated. At other times, it involved more personal aspects of my life. What I had been doing was running my own FB-like procedure, testing and adjusting the parameters until I found something that worked.

It may sound strange, but after I saw how this aspect of the FB procedure helped me muddle through the more stagnant parts of my life, I have become more patient with myself. I find a curious comfort in the realization that, if I can view a setback as FeedBack and keep making adjustments, I will find the right combination of variables sooner or later. Then I can take positive definite action.

Sometimes going nowhere is just what is needed. It provides feedback and an opportunity for adjustment. But you must supply the patience, persistence, and faith.

### FB 100!

\* \* \* \* \*

During the National Educational Computing Conference this past June, I enjoyed asking people about their favorite Logo commands. The variety of responses was wonderful. I especially enjoyed learning the reasons behind their favorite command choices.

I would like to hear from even more people. Do you have a favorite Logo command? Why is it your favorite? How do you feel about it? (I'm talking affective here!) I would be delighted to hear from you. Just send me your name, hometown, favorite Logo command, and why. I hope to incorporate all responses into a future article, just for fun. Many thanks.

P.S.: I would be particularly happy to hear from your students!

*Tom Lough, PO Box 5341, Charlottesville, VA 22905*

## Little Kids and Logo

### A Bushel and a Peck

by Leslie F. Thyberg

Welcome back to school! As a former classroom teacher I can recall the zeal with which my colleagues and I grabbed up any and all ideas for our kids that we could. It is my intention to write a monthly column for you that is chock full of practical, readily useable ideas to use with younger age elementary students. I will try to take into account the broad range of environments that each of your classroom settings represent. Hopefully, the ideas will be adaptable such that whether you have one computer and thirty kids, a lab situation or are blessed with an abundance of computers, you will find something of immediate value. My orientation toward classroom learning is that of an integrated or thematic learning environment. From my perspective this is much more consistent with the kind of informal learning that our young pupils have engaged in prior to their arrival at school. Now that you know some of my biases, let's get on with the show!

### Apple-ology without Apples

This month's column features **bushels** of learning fun. What could be better than a thematic study of apples? Apple-ology has cross-curricular applicability in areas ranging from math to reading to social studies with applications that are as easy as pie. September is a fruitful time of year. Students are refreshed from their summer vacations and are **ripe** for learning. Note: These apple-happy-tasks do not require an Apple computer!

Your activity could begin with a trip to a local orchard or cider mill. Prior to the field trip, be certain to work with your students on vocabulary development. A learning center can be set up that has task cards asking such questions as "What is the difference between an orchard and a forest?" Kids can compile apple-shaped booklets made from construction paper or on computer generated apples using the pattern given later in this article. Depending on computer availability, you might choose to have the whole class construct a chart story describing the field trip experience, or have each student compose a story or story part. At a sharing or group time, the chart story developed by the class (or by teams if you have used small groups) can then be read. Discussions can be had based on the vocabulary words or by singling out specific parts of speech. Better yet, take dictation from the students while seated at the computer. It will give them a chance to see your modelling the use of the computer as a word processor. You can use any word processing

package for this, or LogoWriter, or even straightforward print statements entered in the Logo editor.

## Make an Orchard

No orchard? Make one! How? There are a variety of ways you can create an orchard. One is by using play doh or clay. Some twigs and other natural tidbits can be added to create a realistic effect. Children can be divided into teams. Give the groups the names of different kinds of apples. During the opening weeks of school, these same groups can be the core that you use for rotating them on and off computer or in and out of learning stations. Another idea is to gather some larger branches and plant them in sand inside coffee cans. The construction paper or computer generated apples can then be hung from the trees. Just as there are varieties of apples, so too you can have different trees. For example, one might be a poet-tree on which student generated poetry (about apples or the season) can be hung.

Another variety of tree might be reflective of students; artis-trees in which they experiment with still-life drawings in the style of such greats as Van Gogh and Cezanne. Other possible artis-trees include such off computer activities as making apple prints (much the same one does for potato prints) or apple batik, which is the combined use of bright crayons or oil pastels on manila construction paper, that are washed with tempera paint. After coloring the paper, it can be crunched up or crinkled for a better batik effect. After the tempera wash and the outline of an apple shape can be mounted around it. On computer artistry can be as open ended as you choose to make it. Have students design their own apples or gardens of eatin'. If you prefer, use the following Apple program to allow the kids to branch out on their own. As individuals or in teams, the children can illustrate its interior. The objective, of course, is to allow each student's creativity to blossom.

```
TO LEAF
  SETPC 2
  PU
  SETPOS [0 60]
  SETPOS [5 70]
  SETPOS [10 77]
  SETPOS [15 80]
  SETPOS [20 82]
  SETPOS [25 90]
  SETPOS [25 80]
  SETPOS [22 75]
  SETPOS [20 70]
  SETPOS [15 65]
  SETPOS [10 60]
  SETPOS [0 60]
END
```

```
TO APPLE
  PU
  SETPOS [0 50]
  PD
  SETPOS [10 52]
  SETPOS [20 55]
  SETPOS [30 57]
  SETPOS [40 55]
  SETPOS [50 50]
  SETPOS [60 40]
  SETPOS [55 35]
  SETPOS [50 30]
  SETPOS [47 25]
  SETPOS [46 20]
  SETPOS [47 15]
  SETPOS [50 10]
  SETPOS [55 5]
  SETPOS [60 0]
  SETPOS [64 -10]
  SETPOS [63 -20]
  SETPOS [62 -30]
  SETPOS [58 -40]
  SETPOS [50 -53]
  SETPOS [40 -61]
  SETPOS [30 -60]
  SETPOS [20 -59]
  SETPOS [10 -57]
  SETPOS [0 -57]
  SETPOS [-10 -59]
  SETPOS [-20 -60]
  SETPOS [-30 -61]
  SETPOS [-40 -60]
  SETPOS [-50 -50]
  SETPOS [-55 -40]
  SETPOS [-60 -30]
  SETPOS [-62 -20]
  SETPOS [-65 -10]
  SETPOS [-67 0]
  SETPOS [-67 10]
  SETPOS [-65 20]
  SETPOS [-62 30]
  SETPOS [-57 40]
  SETPOS [-50 50]
  SETPOS [-40 52]
  SETPOS [-30 55]
  SETPOS [-20 55]
  SETPOS [-10 52]
  SETPOS [0 50]
END
```

No cider mill? Pour some! Marvelous math opportunities can develop from learning about measurement and volume using cups, pints, quarts and gallons. Open a cider stand. Maybe your students can actually make some money and with your juicy profits you can purchase some Logo related support materials.

In another part of the orchard you might choose to cultivate a golden rules tree. This is a tree that can grow and develop as the school year progresses. Initially, you might want to hold a class discussion wherein you and the students

## Logo for Little Kids -- continued

establish the rules for using the computer equipment. Golden Apple Rules might be what you choose to hang from the tree. Of course, my favorite golden rule is "Ask three before me." Or, you could make it a primitive tree which displays the most frequently used commands and their abbreviations. These are particularly good ideas for first-time users. Creating a constructive and collaborative environment is critical to the success of your program. Next month's column will include some management techniques and strategies for a successful Logo program in your classroom.

### Apple-ology and Logo

Still another approach to the study of apple-ology is to hang apples or discovery task cards that are Logo and computer related on a tree that can serve as the Learning Tree (a learning center.) You might choose to have a series of precomputer readiness activities which will help familiarize your students with Logo and the particular machines you are using. Or, if you prefer hands-on activities, you can create a set of open-ended discovery activities (original or borrowed from one of the many Logo guides now available.) For example, you could have an apple task card or set of cards with different components of an apple tool kit posted on them. Such a kit can be teacher or student generated. Students can then visit a microworld of apples. Remember the poet-tree? You could lead the class during a sharing session or during the reading and language arts period in a discussion of the vocabulary words or learning experiences gained from the field trip. A simple discussion of parts of speech can lead to the construction of the tool kit. Ask the children to list all the different kinds of nouns (objects, persons, places, things) and related actions that they saw at the orchard (or that are related to the production of cider.) For example

```
TO ORCHARD
OUTPUT PICK [[APPLE TREES WERE RIPE.]
[APPLES TASTE GOOD.]
[ORCHARDS LOOK WONDERFUL.]]
END
```

Next ask for a list of adjectives that describe the objects.

```
TO ADJECTIVES
OUTPUT PICK [CRUNCHY JUICY CRISP TART
RED DELICIOUS]
END

TO FIELD.TRIP
OUTPUT (SENTENCE ADJECTIVES ORCHARD)
END
```

The superprocedure becomes a part of your apple tool kit. You will have to enter a utility for the children to make it run.

```
TO PICK :LIST
OUTPUT
ITEM 1 + RANDOM COUNT :LIST :LIST
END
```

Test the results by typing

```
PRINT FIELD.TRIP
```

several times. Some possible results may be

```
CRUNCHY APPLES TASTE GOOD.
RED APPLE TREES WERE RIPE.
```

Different descriptors can be used. Several simple social studies examples would be to change the lists to describe the equipment used for harvesting (ladders, bushels) or all of different persons involved in the growing, harvesting, marketing, and consumption of apples. Still another idea would be to list kinds of apple juice available on the market and conduct taste tests with consumer responses. Science can be woven in, by extending the vocabulary discussion to include terms such as graft, pistil, and mulch.

A science classroom in Tarentum, Pennsylvania did a project involving the five seasons of the apple. Four were the seasons of the calendar year. The fifth was when the apples are sent to market and eaten! Apple math might take place in a learning center where you have placed a bushel or bucket of apples. Students can estimate the number of apples or how much they weigh. Better yet, the class can make applesauce or caramel apples (or if you are very brave, apple pie.)

Apple-ology can be a bushel of fun. Good luck!

### Resources:

- Gibbons, G. (1984) *The seasons of Arnold's apple tree*. San Diego: Harcourt, Brace, Jovanovich.
- Green, C. and Jaeger, C. (1984) *Teacher, kids, and Logo*. Irvine CA: EduComp Publications.
- Hogrogian, Nonny. *Apples*. New York, MacMillan.
- Hunt, B. (1976) *Apples: A bushel of fun and facts*. New York: Parents Magazine Press.
- Johnson, H. (1977) *From apple seed to applesauce*. New York: Lothrop.
- Scheer, J. (1964) *The rain makes applesauce*. New York: Holiday House.

I welcome your correspondence and comments. This column is, after all, for you. If you have a neat idea, sent it to me. I'll cite you and you can brag to all of your friends that you are now famous and widely read!

Dr. Leslie F. Thyberg, 5637 Rippey Street,  
Pittsburgh, PA 15206, AppleLink ALS 038



## Logo Ideas

### (Almost) all you need to know about subdirectories in LogoWriter 2.0

by Eadie Adamson

Subdirectories are nothing more than "organizers" for your files. Just as you might sort your cancelled checks into groups (medical, business, publications, educational expense, and so on) you can subdivide the files on your disk into appropriate groupings.

If you have a multi-page project, think of a subdirectory as a project organizer. The subdirectory will contain all the shapes and pages for that particular project. Each subdirectory in LogoWriter 2.0 has its own Shapes page. This means you can have many shape pages on a single disk. Each page in a subdirectory uses the shapes for that subdirectory. (GETSHAPES loads the Shapes page quickly, without leaving the page you are on.)

At first, subdirectories seem quite mysterious. The process of creating them is very "un-Logo-like." Just follow the steps carefully and in no time you will be quite comfortable with creating and using subdirectories.

#### Creating Your First Subdirectory

Choose NEWPAGE and then type

```
CREATEDIR "NAME.OF.SUBDIRECTORY
(Use whatever name you want)
```

Check to see that that directory was created by pressing Esc or by typing CONTENTS. On the Contents page below the LogoWriter pages, you should see a blank space, then the name you gave your subdirectory, preceded by %, LogoWriter's substitute for the slash used in ProDos.

```
%NAME.OF.SUBDIRECTORY
```

If you created a new subdirectory, this is all you need to do. Now choose that subdirectory just as you would choose a page. When you press return, instead of a new page, you will see the contents of that subdirectory. Note that the Contents page shows you the directory name, preceded by %, in the upper left corner. You may edit your shapes in that subdirectory without affecting the Shapes page on the main contents page.

Choose a new page, name it, and then go back to CONTENTS. Go to the Shapes page and change a shape. Press Esc. Note that each time you pressed Esc you returned

to the contents of the subdirectory. You need to press Esc again to get back to the main Contents page. (Erase the page as usual, if you don't want to keep it, with Apple - 6.) Check the Shapes page on the main directory. You will see that the change you made in the subdirectory has not affected the main Shapes page.

#### Creating a Subdirectory from Previously Created Work Using a Single Disk Drive

When you first think about organizing subdirectories, take a 2.0 disk on which you already have a number of files which you wish to organize in subdirectories. Start by using only a single drive. Put your disk in the first drive, the primary drive.

First, create the subdirectory, as outlined above. Then follow these instructions to copy the pages into the subdirectory:

Choose NEWPAGE from the main Contents page on your old disk.

First, be sure you know the "name" or "prefix" of your disk. (Every ProDos disk has a name; in ProDos it is preceded by a /; in LogoWriter it is preceded by a %.) Type

```
SHOW ONLINE
```

You will see a list that includes the name of your disk, e.g., [%MY.DISK%]

Get a list of your pages. Type

```
PRINT PAGELIST
```

(PRINT puts the PAGELIST on the page; SHOW places it in the COMMAND CENTER.)

Decide which of these pages are to go in the new subdirectory.

You are now ready to copy the pages one at a time into the subdirectory, using the following command line as a pattern:

```
COPYFILE "%MY.DISK%PAGENAME
"%MY.DISK%NEW.SUBDIR.NAME%PAGENAME
```

COPYFILE is the command to copy a file (a LogoWriter page). Inputs for COPYFILE are as follows:

After the first "%:

the first input: the name of the disk (which you got by typing SHOW ONLINE).

the second input: the name of the page to copy.

## Logo Ideas -- continued

After the second "%:

- the first input: the name of the disk (which you got by typing SHOW ONLINE).
- the second input: the name of the new subdirectory you just created.
- the third input: the name of the page you want to copy into the subdirectory.

The PAGENAME in the new subdirectory can have any name you wish; it does not have to be the same name as the page you are copying. When you press Return, the drive will go on and the copying will take place.

When your cursor returns, you can delete the pagename from the command line and substitute the next page (rather than retyping the entire COPYFILE "%MY.DISK, etc. each time). Try using Control-D or Delete to delete the first pagename, Apple-6 to delete the last one.

When you are finished copying the files, press Esc and choose your new subdirectory from the contents page. (Type LEAVEPAGE so you don't have to name your page.) Check to be sure you copied all the pages. If you wish, return to the main directory and erase the pages which you have copied into the new subdirectory (you don't need two copies on a single disk).

### Using Two Drives Copying from One Disk to Another

First, create anew subdirectory on the disk. Be sure you are working on the correct drive. It is less confusing if you place the disk to copy from in A, the first disk drive. Place the disk to copy to in B, the second drive. Move from drive to drive by typing:

```
SETDISK "A (for Drive 1)
```

and

```
SETDISK "B (for Drive 2)
```

First, move to the drive holding the disk on which you want the new subdirectory (Drive 2).

Type:

```
SETDISK "B
CREATEDIR "NEWDIR
(notice there is no percent sign)
```

Find out the the names of the disks in the two drives:

```
SHOW ONLINE
```

Now follow this pattern in which the first set of inputs is the disk copied from, the second is the disk copied to (you can do this from either drive since you are specifying the "from" and the "to" by giving the prefix each time):

```
COPYFILE "%FROM%SUBDIR%PAGENAME
"%TO%SUBDIR%PAGENAME
```

to copy from a disk with a subdirectory to another disk with a subdirectory, or

```
COPYFILE "%FROM%PAGENAME
"%TO%SUBDIR%PAGENAME
```

to copy from a disk with no subdirectories into a subdirectory on the other disk.

The inputs required are the following:

First set:

```
"%DISK FROM
—the disk name you are copying from
%SUBDIRECTORY—the subdirectory name you
are copying from, if any
%PAGENAME—the name of the page to copy
```

Second set:

```
"%DISK TO—the disk name you are copying to
%SUBDIRECTORY—the subdirectory name you
are copying to, if any
%PAGENAME—the name of the page in the new
subdirectory
```

### Copying an Entire Subdirectory to Another Disk

Create the subdirectory on the second disk, placed in the second drive, drive B.

Move back to the first drive, drive A:

```
SETDISK "B
NEWPAGE
CREATEDIR "NEWSUBDIR
SETDISK "A
```

Go to the subdirectory you want to copy from. Choose NEWPAGE in the subdirectory.

```
PRINT PAGELIST
```

Use this pattern (type as one line, although it will wrap to a second line):

```
COPYFILE
"%DISK.FROM%SUBDIR.FROM%PAGENAME
"%DISK.TO%SUBDIR.TO%PAGENAME
```

Be sure to copy the Shapes page also.

## Using the Copy Tools

You can also use the COPY.TOOLS procedure COPYEH to copy a subdirectory from one disk to another (this will copy all files in the subdirectory). It takes fewer steps once the procedure is loaded.

Insert a disk with COPY.TOOLS in the first drive. Choose a NEWPAGE (not the COPY.TOOLS page!). Type:

```
GETTOOLS "COPY.TOOLS
```

Press Esc to go to the Contents page. Insert your disk with the subdirectory, press Esc, and choose a NEWPAGE again.

First, create the subdirectory on the disk you are transferring to, in the second drive. (Use SETDISK "A and SETDISK "B to move between drives.)

```
SETDISK "B
CREATEDIR "NEWSUBDIR
COPYEH "%DISK.FROM%SUBDIR
"%DISK.TO%SUBDIR
      (from either drive)
```

In this case you do not need pagenames, since COPYEH takes care of that. This really works well, but I often find it easier to do the copying manually rather than getting tools from another disk if I only need to copy a page or two. Suit yourself! After you have gone through the process a number of times it will not seem so complex.

### Note:

If you are copying shapes into an already created directory, you may encounter a problem. If you have used the Shapes page in that subdirectory, then you must first erase the shapes from that directory:

```
ERASEFILE "%MY.DISK%SUBDIR%SHAPES
```

Then you can copy the Shapes page from any other directory, including the main directory, into that subdirectory.

```
COPYFILE "%MY.DISK%SHAPES
"%MY.DISK%SUBDIR%SHAPES
```

**The West Coast Logo  
and  
Telecommunications Conference**

**February 3 - 4, 1989  
Los Angeles Hilton Hotel**

## QUICK SUMMARY FOR SUBDIRECTORIES

### MAKING A NEW SUBDIRECTORY AND COPYING PAGES TO IT

```
CREATEDIR "new subdirectory
SHOW ONLINE
      (gets the prefix of your disk)
COPYFILE "%MYDISK%PAGENAME
"%MYDISK%newsubdirectory%PAGENAME
```

### COPYING FROM ONE DISK TO ANOTHER

Place the second disk, the disk to copy TO, in the second drive:

```
SETDISK "B
CREATEDIR "NEWDIR
```

Move back to the first drive:

```
SETDISK "A
PRINT ONLINE
PRINT PAGELIST
```

```
COPYFILE "%MYDISK%PAGENAME
"%NEWDISK%NEWDIR%PAGENAME
```

If necessary, also copy the Shapes page:

```
COPYFILE "%MYDISK%SHAPES
"%NEWDISK%NEWDIR%SHAPES
```

If you get a message that there is already a page named SHAPES, use the command:

```
ERASEFILE "%NEWDISK%NEWDIR%SHAPES
```

Then copy the Shapes page.

```
COPYFILE "%MYDISK%SHAPES
"%NEWDISK%NEWDIR%SHAPES
```

### QUICK COPY OF AN ENTIRE SUBDIRECTORY

```
SETDISK "B
CREATEDIR "NEWSUBDIR
```

Load COPY.TOOLS as tool pages by inserting a disk with the COPY.TOOLS page in the first drive. On a NEWPAGE type:

**Logo Ideas -- CONTINUED**

```
SETDISK "A
GETTOOLS "COPY.TOOLS
```

Return to CONTENTS. Insert the disk with the subdirectory to be copied from in Drive A, choose a NEWPAGE:

```
SETDISK "A
SHOW ONLINE
COPYEH "%MYDISK%SUBDIR
"%NEWDISK%NEWSUBDIR
```

**A Footnote for Subdirectories****Neat Idea!**

Susan Hansen, a colleague of mine, was converting her second grade disks to 2.0. "Wouldn't it be neat to put all their old work in a subdirectory? Do you think it would work?" she asked.

My first response was to try the COPY.TOOLS. COPYEH turns out to be rather schizophrenic when it comes to putting pages on the same disk into a subdirectory on the same disk. Like the "Sorcerer's Apprentice" (remember the cartoon version with Mickey Mouse and millions of brooms?) COPYEH does the job once, then creates another subdirectory, copies again, and on and on until the disk is full! Our solution was to watch for the second round of copying, stop it, then erase the extra files and the extra subdirectory. There had to be an easier way!

I went home and wrote these procedures which put the pages into a subdirectory and then erase the pages outside the subdirectory afterwards. To use my program, simply type COPYTOSUB and press Return. You will be asked for the name of the subdirectory. The procedures listed below will then create the subdirectory, copy the pages into the subdirectory, tell you when the work is complete, and give you the option of having the now extraneous pages outside the subdirectory erased for you. One caution: if you do this, be sure to clear the disk of useless pages first. Remember that a subdirectory does take up space! Here are the procedures which organized Susan's student disks neatly and easily:

**COPY TO SUBDIRECTORY: SAME DISK, ALL FILES**

```
TO COPYTOSUB
CT
CC
TYPE [What name for the subdirectory?\ ]
```

```
NAME FIRST READLISTCC "SUBDIR
NAME FILELIST "FILES
TYPE CHAR 13
TYPE (SENTENCE [The subdirectory]
WORD "% :SUBDIR
[will now be created on this disk.])
TYPE CHAR 13
CREATEDIR :SUBDIR
COPYALL :FILES PREFIX
(WORD PREFIX :SUBDIR "%)
END
```

```
TO COPYALL :FILES :FROMPATH :TOPATH
IF EMPTY? :FILES [PRINT (SENTENCE
[COPIING TO]:SUBDIR "COMPLETE)
ERPAGES STOP]
PRINT (SENTENCE [COPIING] FIRST FIRST
:FILES [FROM] :FROMPATH [TO]
:TOPATH)
COPYFILE (WORD :FROMPATH FIRST FIRST
:FILES) (WORD :TOPATH FIRST
FIRST :FILES)
COPYALL BUTFIRST :FILES :FROMPATH :TOPATH
END
```

```
TO ERPAGES
CHECK
END

TO CHECK
PRINT [You are about to erase all pages
EXCEPT those in the subdirectory!!!]
PRINT "
PRINT [ARE YOU SURE YOU WANT TO DO THIS?]
NAME READLISTCC "ANS
IFELSE MEMBER? "Y :ANS [DOIT][PRINT "
PRINT [OKAY, pages will NOT be erased.]]
END
```

```
TO DOIT
NAME PAGELIST "PAGES
ERPS :PAGES
END
```

```
TO ERPS :PAGES
IF EMPTY? :PAGES [PRINT [WORK COMPLETE]
STOPALL]
ERPAGE FIRST :PAGES
ERPS BUTFIRST :PAGES
END
```

[Eadie Adamson, Allen Stevenson School, 132 East  
78th Street, New York, New York 10021]

## Stager's Stuff

### Another Bunch of Election Stuff

by Gary S. Stager

In April, we explored some of the ways in which Logo can be used as the centerpiece of a curriculum unit on elections. I shared with you a sophisticated voting booth program which can be used in your classroom by you and your students<sup>1</sup>. This month's article will begin with procedures, which when used in conjunction with last month's voting machine, will create pie charts illustrating the election results. The second part of this article will explore a Logo project that allows teachers and students to enter referendum questions, vote on these yes/no issues, tabulate, and graph the results of the polling.

#### Pie Chart Procedures for the Voting Machine

Add the following procedures to the flip-side of your voting machine page. If you have translated last month's project to run in other versions of Logo, then enter these additional procedures using the Logo editor. Remember that due to the size of the voting machine and graphing procedures you need to use a 128K version of Logo (LogoWriter 2.0, IBM Logo, LCSi Logo II, MacLogo, or Object Logo)<sup>2</sup>.

Labeling the pieces of a pie graph from within procedures is a programming project that is a bit to cumbersome for this article. For this reason, I have decided to have Logo print the list of candidates with their corresponding vote counts and overall percentages. If you are using LogoWriter, you can easily label the pie manually from the command center.

After your class has voted you can type TOTALS to see a table of the vote totals by party in descending order. You may also use the new PLOT procedure to graph the results of one party at a time. PLOT is used in the following way:

```
PLOT :DEMOCRATIC
or
PLOT :REPUBLICAN
```

The input to the plot procedure must be one of party lists created by the Voting Machine program in the April LX. In other words, if you told the voting machine program that you want a BLUE party and a RED party, then you would type:

```
PLOT :BLUE
PRINTSCREEN
PLOT :RED
PRINTSCREEN
```

The following quick reference to the procedures you

need to know to easily use the voting machine tools.

**VOTING.MACHINE:** Setup the party and candidate names and cast first vote

**VOTE:** VOTE should be executed once for every voter or REPEAT some number [VOTE]

**TOTALS:** Displays the election results — can be run at any time you wish to see the way the voting is going

**PLOT :party:** Draws a pie chart representing the votes for one party

#### THE ADDITIONAL PIE CHART PROCEDURES

```
TO PLOT :LIST
RG
CT
HT
PU
MAKE "RADIUS 360/PI/2
SETPOS [-50 0] PD
MAKE "TOTAL TOTAL :LIST
CIRCLE
PRINT (SENTENCE "PLOT :LIST)
PRINT []
PLOTTER :LIST
SETH 0
GO.FILL :LIST
SETH 0
END

TO RESULTS :LIST
PRINT (SENTENCE
  FIRST :LIST LAST :LIST TABS "=" WORD
  (ROUND (100 * LAST :LIST) / :TOTAL) "%)
END

TO PI
OUTPUT 3.14159
END

TO CIRCLE
REPEAT 360 [FORWARD 1 RIGHT 1]
RIGHT 90
PU
FORWARD :RADIUS
LEFT 90
END

TO PLOTTER :LIST
IF EMPTY? :LIST [STOP]
SEG (100 * LAST THING UNPARSE FIRST :LIST) / :TOTAL
RESULTS THING UNPARSE FIRST :LIST
PLOTTER BUTFIRST :LIST
END

TO SEG :PERCENT
```



## Stager's Stuff -- continued

```

PD
FILLSEG (:PERCENT / 100) * 360
END

TO TOTAL :LIST
IF EMPTY? :LIST [OUTPUT 0]
OUTPUT SUM (LAST THING UNPARSE FIRST :LIST)
(TOTAL BUTFIRST :LIST)
END

TO COLORER
IF COLOR = 5 [SETC 1 STOP]
SETC COLOR + 1
END

TO FILLSEG :TURN
IF :TURN < 1 [STOP]
REPEAT INT :TURN - 1 [RIGHT 1]
FORWARD :RADIUS
BACK :RADIUS
RIGHT 1
END

TO SUM :N1 :N2
OUTPUT :N1 + :N2
END

TO GO.FILL :LIST
PLOTTER1 :LIST
END

TO FILLER
COLORER
PU
FORWARD 25
PD
FILL
PU
BACK 25
END

TO PLOTTER1 :LIST
TONE 800 10
IF EMPTY? :LIST [STOP]
SEG1 (100 * LAST THING UNPARSE FIRST :LIST) / :TOTAL
PLOTTER1 BUTFIRST :LIST
END

TO SEG1 :PERCENT
PD
FILLSEG1 (:PERCENT / 100) * 360
END

TO FILLSEG1 :TURN
REPEAT (INT :TURN) / 2 [RIGHT 1]
FILLER
REPEAT (INT :TURN) / 2 [RIGHT 1]
END

TO TABS
OUTPUT [ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ ]
END

```

## The Referendum Ballot

Referendum questions are an important and often misunderstood part of the election process. I thought that it would be interesting to create a voting machine that would let you vote on referendum questions (questions with yes or no answers). What I have really created is a survey program which permits students to enter five questions (you can modify the program to allow more questions, but the screen is only large enough to graph five questions), tabulate the results, and create a bar graph of the results which can then be printed<sup>3</sup>. The LogoWriter print materials and the Terrapin Logo Data Toolkit contain an excellent variety of Logo activities for exploring survey taking, interpreting data, and graphing. The referendum project runs in a 64K version of LogoWriter and can be easily modified to run in other versions of Logo.

If you type

STARTUP

you see

**Enter up to 5 ballot questions. When you are done, type: DONE**  
**QUESTION #1** Do you like ice cream?  
**QUESTION #2** Do you have a dog?  
**QUESTION #3** Do you have a cat?  
**QUESTION #4** Should school be open during the summer?  
**QUESTION #5** done

Then, typing

VOTE

gives you:

**Do you like ice cream?**  
 YES  
**Do you have a dog?**  
 NO  
**Do you have a cat?**  
 NO  
**Should school be open during the summer?**  
 NO

You can type

RESULTS

any time you want to see the results so far:

**Do you like ice cream?**  
**YES = 22 NO = 3**

**Do you have a dog?**

**YES = 8 NO = 17**

**Do you have a cat?**

**YES = 5 NO = 20**

**Should school be open during the summer?**

**YES = 0 NO = 25**

By typing GRAPH, you can display the results of the referendum ballots in the form of a bar graph. If the number of yes or no votes for a particular question is greater than 30, a bar 28 units high is drawn and the correct number of votes is labeled above the bar. If a question didn't receive any yes or no votes, no bar will be drawn. I shaded the bars with a texture instead of filling them with color so that a printout would be more dramatic. If your version of Logo does not have SHADE, you may substitute FILL or leave the bar hollow in the FILLER procedure.

## Referendum Procedures

In the command center, type:

NAMEPAGE "REFERENDUM

then enter these procedures on the flip side of the page.

```
TO STARTUP
  CLEARNAMES
  GET.QUESTIONS
END
```

```
TO VOTE
  CT
  CG
  ASK.QUESTIONS :Q.LIST
END
```

```
TO GET.QUESTIONS
  MAKE "Q.LIST []
  QUESTION.INFO
  QUESTIONS 1
END
```

```
TO QUESTION.INFO
  CC
  TYPE (SENTENCE [Enter up to 5 ballot questions. When
    you are done, type: DONE] CHAR 13)
END
```

```
TO QUESTIONS :COUNT
  IF :COUNT > 5 [STOP]
  TYPE SENTENCE "QUESTION (WORD "# :COUNT)
  MAKE QN :COUNT (LIST READLISTCC 0 0)
  IF (FIRST THING QN :COUNT) = [DONE]
    [CLEARNAME (QN :COUNT) STOP]
  MAKE "Q.LIST LPUT (QN :COUNT) :Q.LIST
  QUESTIONS :COUNT + 1
END
```

```
TO ASK.QUESTIONS :LIST
  IF EMPTY? :LIST [STOP]
  PRINT FIRST THING FIRST :LIST
  IFELSE "Y = READCHAR [YES FIRST :LIST] [NO FIRST
    :LIST]
  ASK.QUESTIONS BUTFIRST :LIST
END
```

```
TO YES :QUESTION
  PRINT "YES
  MAKE :QUESTION (LIST FIRST THING :QUESTION (FIRST
    BUTFIRST THING :QUESTION) + 1 LAST THING :QUESTION)
END
```

```
TO NO :QUESTION
  PRINT "NO
  MAKE :QUESTION (LIST FIRST THING :QUESTION
    (FIRST BUTFIRST THING :QUESTION)
    (LAST THING :QUESTION) + 1)
END
```

```
TO RESULTS
  CT
  TOTALS :Q.LIST
END
```

```
TO TOTALS :LIST
  IF EMPTY? :LIST [STOP]
  PRINT FIRST THING FIRST :LIST
  PRINT (SENTENCE
    "YES "= FIRST BUTFIRST THING FIRST :LIST
    "NO "= LAST THING FIRST :LIST)
  PRINT []
  TOTALS BUTFIRST :LIST
END
```

```
TO QN :COUNT
  OUTPUT WORD "QUESTION. :COUNT
END
```

## Referendum Graphing Procedures

```
TO GRAPH
  GRID
  GRAPHER :Q.LIST
END
```

```
TO GRID
  RG
  HT
  CT
  PU
  SETPOS [-110 -75]
  PD
  NOTCHES 0
  BACK 160
  RIGHT 90
  FORWARD 240
  PU
  SETPOS [-95 85]
  SETH 90
  MARK 1
  SETPOS [-100 -85]
  REPEAT 5 [LABEL [Yes No] FORWARD 50]
  PU
  SETPOS [-100 -75]
```

# Stager's Stuff -- continued

```

SETH 0
END

TO NOTCHES :COUNT
IF :COUNT > 30 [STOP]
PD
RIGHT 90
FORWARD 5
BACK 10
PU
BACK 15
LABEL :COUNT
FORWARD 20
LEFT 90
PD
FORWARD 10
NOTCHES :COUNT + 2
END

TO GRAPHER :LIST
IF EMPTY? :LIST [STOP]
SETC 1
BAR1 FIRST BUTFIRST THING FIRST :LIST
SETC 1
MOVE 20
BAR2 LAST THING FIRST :LIST
MOVE 30
GRAPHER BUTFIRST :LIST
END

TO BAR1 :HEIGHT
IF :HEIGHT > 30 [SETC 4 BAR1 28 PU FORWARD 150
  LABEL :HEIGHT BACK 150 STOP]
IF :HEIGHT = 0 [MOVE 20 STOP]
SETSH 29
SETH 0
PD
REPEAT 2 [FORWARD :HEIGHT * 5 RIGHT 90
  FORWARD 15 RIGHT 90]
FILLER
END

TO BAR2 :HEIGHT
IF :HEIGHT > 30 [SETC 4 BAR2 28 PU FORWARD 150
  LABEL :HEIGHT BACK 150 STOP]
IF :HEIGHT = 0 [MOVE 30 STOP]
SETSH 30
SETH 0
PD
REPEAT 2 [FORWARD :HEIGHT * 5 RIGHT 90
  FORWARD 15 RIGHT 90]
FILLER
END

TO FILLER
PU
RIGHT 35
FORWARD 7
PD
SHADE
PU
BACK 7
LEFT 35
END

```

```

TO MARK :COUNT
IF :COUNT > 5 [STOP]
LABEL WORD "# :COUNT
FORWARD 50
MARK :COUNT + 1
END

```

```

TO MOVE :HOW.FAR
SETH 90
FORWARD :HOW.FAR
SETH 0
END

```

Although the procedures given in April and this month's column feature some nice examples of list processing, my purpose for writing them was to provide teachers and students with Logo tools they could use to get caught up in the democratic process. I would be very interested in hearing your reactions to these project ideas and would love to see what other types of "election stuff" you and your students may have created.

Hope you had a great summer! Did you take a Logo course, create a Logo project that is interesting to you, attend Dan and Molly Watt's Lesley College Logo Institute, get your modem working, re-read *Mindstorms*. Now it's time to apply all that you learned to a new school year. I hope you'll be delighted with the neat stuff that popped into my head during the summer!

## Notes:

1. Jim Muller - Sysop of the CompuServe LogoForum has generously translated last month's voting machine program into IBM Logo. Both the LogoWriter 2.0 version and the IBM Logo version are available in Data Library 6 of the Logo Forum. Type GO LOGOFORUM at any CompuServe Prompt.
2. I will upload a scaled-down 64K version of the voting machine, a general purpose pie chart program, and this article to the Logo Forum on CompuServe. If you don't feel like typing, you can simply (well not too simply) download these programs from CompuServe.
3. In versions of Logo other than Apple LogoWriter, you may need to adjust the screen coordinates (SETPOS) to fit your screen dimensions.

*Gary Stager is the Director of Training for the Network for Action in Microcomputer Education. He can be reached at: 5 Eastside Avenue, Wanaque, NJ 07465; ph: 201-633-3121; CIS: 73306,2446; Applelink: K0331.*

## Logo LinX

### Secular Conversions

by Judi Harris

How many times has some computer science-type tried to convince you that Logo is "an inferior graphics language?" More times than you have fingers and toes? Perhaps you find it as difficult as I do to explain that, although Logo is often used in the graphics mode, its purposes are far from being the next MacPaint competitor.

Your students, no doubt, have long since realized that Logo is not the first-string choice for effecting graphic realism. But, now and again, identifiable pictures ("Ah, yes! A maple tree!") are needed for use in Logo programs. Somehow, stick figures can bely the sophistication of the code by which they appear, not to mention that a turtled drawing of even moderate detail can take eons to manifest in full geometric splendor.

To these ends, I offer this month's "Logo LinX."

#### Means or Ends?

In education, we are fond of saying that it makes no sense to reinvent the wheel if what we intend to do is roll. Turtle drawings can be by-products of rich geometric exploration, but when pictures are needed as tools for Logo text investigation, they should be imported, not invented.

Perhaps you have already assembled a Logo picture library for this purpose from procedures and screens that your students have generated. It is also possible to use graphics originally designed for use with Print Shop, a popular signmaking program, in your Logo procedures. In a similar manner, Logo-generated pictures can be used as Print Shop graphics.

Consider, for example, the excited fifth-grader that wants to make a Logo alphabet program for a kindergarten class. She would like the computer to display a picture of an object beginning with whatever letter the user presses. She is not satisfied with the size and limited selection of memory-resident turtle shapes. This programming challenge is appropriate for someone investigating list processing structures, but the time required to create 26 different graphics is probably more than she can schedule in her school's computer lab. She may abandon the idea for practical reasons.

Or, consider the fifth-grade Transformer aficionado who has spent the last two months carefully coding a detailed diagram of his favorite robot. He would now like to use this picture to make invitations to his birthday party. The party is in two weeks, and he needs more time than that to learn

enough list processing to generate the invitations with Logo. It seems logical to him that Print Shop should be able to use his picture, since he used a computer to create the image.

You could speak to these two students about operating system differences, picture packing routines, and economic advantages for software corporations that design incompatible software, but chances are that they will only remember the disappointment. Or, you could help them to roll by providing a wheel that has already been invented.

#### To Pack or Not to Pack?

Logo graphics saved to disk with the SAVEPIC command exist as full screen, high-resolution binary picture files. Print Shop graphics are also stored as binary files, but are much smaller in size because they have been packed, or coded in such a way as to save disk space. User-created graphics to be used with Print Shop must be packed according to the same system. This is the first reason why Logo pictures, which are completely UNpacked in their full-screen forms, cannot be used as Print Shop graphics.

(There is one notable exception to this rule. The newer versions of Print Shop, such as Color Print Shop, accept "screens" as input under the "Screen Magic" option. This is an easy way to obtain color printouts of Logo pictures with color printers such as the Imagewriter II.)

Most recent versions of Logo for the Apple II computer family are written in ProDOS, or Professional Disk Operating System. (The exception is LogoWriter Version 1.0 and 1.1 which does not use either DOS or ProDOS.) Most versions of Print Shop are written in DOS 3.0-3.3, older Apple disk operating systems which are unfortunately directly incompatible with ProDOS programs and data files. Fortunately, Apple provided a ProDOS User's Disk when it released ProDOS several years ago, which contains a utility that will convert ProDOS files to DOS format, and visa-versa.

Yet a ProDOS Logo picture that has been converted to DOS 3.3 format with the ProDOS User's Disk still cannot be used as a Print Shop graphic, and Print Shop graphics converted to ProDOS cannot be used in Logo procedures for reasons that you probably have already guessed. Logo requires unpacked high-resolution graphics; Print Shop requires specially packed graphic code.

Enter the Beagle Brothers, an exceptional software firm noted for its dry wit, on-your-honor software copyright policy, and powerful utility programs. Beagle Brothers manufactures three inexpensive (\$19.95) collections of Print Shop graphics that they call Minipix. More appropriate for

## Logo LinX -- continued

our purposes, though, are the editing tools that they include with each disk. The Minipix editor allows the user to convert hi-res screens to Print Shop format, and packed Print Shop pictures to hi-res images that can be used with Logo. The Beagle tools also offer options for picture proportion changes, text addition, horizontal and vertical inversions, and negative image generation.

### Ready to Cavort (Convert)?

Instructions for these "secular conversions" follow. A minimum of six disks are needed to proceed: Print Shop (DOS), Minipix #1, #2, or #3 (DOS), Logo (ProDOS or DOS), a ProDOS User's Disk, a DOS files diskette, and a ProDOS files diskette.

#### To Use Print Shop Pictures with Logo

1. Load a Beagle Bros. Minipix Disk, side 1.
2. Press "E" to choose the "Editor" option.
3. Put the disk with Print Shop pictures in drive 2 (if available,) or remove the Beagle Bros. disk from drive 1, and replace it with the Print Shop pictures disk.
4. Press <RETURN> when prompted to do so.
5. If the pictures disk is in drive 2, press "D" to change the drive number to 2.
6. Press "L" to "Load a Minipic." This means that the computer will load a packed Print Shop picture into its memory.
7. Type the number of the box where you would like to place the picture. (Use a number between 1 and 4.)
8. Type the name of the picture that you want to load and press <RETURN>.

NOTE: If you need a list of pictures on the disk, type "?". Press the space bar to proceed through the catalog, and <RETURN> at the end of the catalog.

Load up to four Print Shop pictures into the computer's memory in this way.

9. Now set the size proportions of the Print Shop pictures as you want them to appear in your Logo program. To do this, type "B" and the number of the box where the picture to "blowup" is displayed, i.e., B1 or B2.
10. Keep pressing "C" until the picture is displayed in satisfactory size and proportions.
11. Press "S" to change to the screen mode.

12. Options "F" (to flip the image vertically), "M" (to mirror the image horizontally), and "N" (to change the image to its corresponding negative) are now available for use. Make any desired changes by typing the appropriate letters and pressing <RETURN>.
13. Replace the Print Shop pictures disk with your DOS 3.3 data disk.
14. Press "S" to save the picture in a format that can be used with Logo. You will be asked to type a picture name, and then press <RETURN>.

NOTE: If you plan to use this picture with Terrapin or Krell Logo, end the picture name with .PICT. The picture is now ready to load into Terrapin or Krell Logo with a READPICT "(name) command. DON'T type the .PICT at the end when you load the picture.

**\*\* Two disk drives are needed to complete steps 15 to 25. \*\***

15. If you are going to use this picture with an LCSII ProDOS Logo (LCSII Logo II, Apple Logo II, LogoWriter Version 2.0), put the ProDOS User's disk in drive 1, and reboot the system.
16. Press "C" for DOS—>ProDOS conversion.
17. Put the DOS 3.3 data disk in drive #2, and the ProDOS data disk in drive #1.
18. Press "P" and then "S" to set the pathname.
19. Press <RETURN> twice to set pathname by slot and drive.
20. Press "T" to transfer files, then press <RETURN> for a list of files. The screen will display a catalog of file names on the disk.
21. Use the up and down arrow keys to move the highlighting to the picture(s) that you would like to use with Logo. When each selection is highlighted, press the space bar.
22. Press <RETURN> to complete the conversion process.
23. Boot the Logo language disk, and replace it with the ProDOS data disk.
24. Type LOADPIC "picture name and press <RETURN>.
25. The picture will appear in fullscreen mode. Type SPLITSCREEN or SS <RETURN> to display the command lines.



## To Use Logo Pictures with Print Shop

### Logo Pictures as Hi-Res Screens:

1. Boot the system with Color Print Shop.
2. Select "Screen Magic" from the main menu.
3. Select "Get Screen" from the Screen Magic menu.
4. Insert the data disk containing the Logo picture that you wish to use, type the full name of the picture file when prompted to do so, and press <RETURN>.

NOTE: If you are using a Logo picture file generated with Terrapin or Krell Logo, the file name ends with ".PICT". This extension must be typed, i.e., TURTLE.PICT.

5. If you wish to add text to the picture, select "Draw Text on Screen."
  - a. Select "Screen in Memory" from the submenu.
  - b. Select a font for the text letters.
  - c. Type the text, remembering where the Logo graphic is situated on the hi-res screen.
  - d. Choose "No" when prompted with the option to center the message.
6. The text will be displayed with the Logo graphic. If the placement is satisfactory, select "Save Screen" and enter a name when prompted to do so.

### Logo Pictures as Print Shop Graphics or Beagle Bros. Minipix:

1. Save the Logo graphic as a picture with the LCSi SAVEPIC "title command.

NOTE: Pictures originally created with Terrapin or Krell Logo are not readable by the Minipix editor, even after they have been converted twice (DOS—>ProDOS, then ProDOS—>DOS) with the ProDOS User's Disk.

2. Convert this ProDOS picture to DOS 3.3 format by following steps 15 through 22 under "To Use Print Shop Pictures with Logo," with one addition. Before completing step 18, press "R" to reverse the conversion path (ProDOS to DOS 3.3).
3. Load a Beagle Bros. Minipix Disk, side 1.
4. Press "E" to choose the "Editor" option.
5. Press <RETURN> when prompted to do so.
6. Put the disk with the converted DOS 3.3 picture(s) on it in drive 2.

7. Press "D" to change the drive number to 2.
8. Press "X" to change to the hi-res screen mode.
9. Press "L" to "Load a Screen." This means that the computer will load a converted Logo picture into its memory.
10. Type the full file name of the picture to be loaded, then press <RETURN>.
11. Press the space bar at any time to see the full screen image of the Logo picture. Press it again to see the available Minipix commands.
12. Now that the Logo picture is in the computer's memory, the Minipix program will help you to convert it to a packed form that is usable with Print Shop's banner, sign, letterhead, and greeting card options.

Type "C" to capture part of the screen image into a Minipix box, then type the number of the box to use. A maximum of 4 screen portions can be captured in this way.

13. Remember that the picture will be converted exactly as it appears on the screen. Most Logo pictures appear with a black background and light-colored lines. Press "N" to reverse to the negative image, which will print on paper as black lines on a white background, and therefore conserve printer ribbon.
14. Move the box outline that appears according to the directions on the screen. Press <ESC> when finished.
15. Press "S" to save the minipic(s) on the DOS 3.3 files diskette, and type a name for the file when prompted to do so, followed by <RETURN>. (Notice that this type of file occupies 4 disk units when the files diskette is CATALOGed, as opposed to a hi-res screen file, which occupies 34 disk units.)
16. This "Logo Minipic" may now be used when constructing a Print Shop banner, sign, greeting card or letterhead. Enter the DOS 3.3 file name chosen in the last step when Print Shop prompts you to "Select a Graphic."

\*\*\* The author wishes to acknowledge and thank Chris Appert of the Kluge Rehabilitation Center in Charlottesville, VA for collaborating on the first versions of these instruc-

*Judi Harris, 621F Madison Avenue, Charlottesville, VA 22903; CIS: 75116,1207; BitNet: jbh7c@Virginia*

## MathWorlds

edited by  
A. J. (Sandy) Dawson

"...a culture is constantly in process of being recreated as it is interpreted and renegotiated by its members. In this view, a culture is as much a forum for negotiating and re-negotiating meaning and for explicating action as it is a set of rules or specifications for action.

Education is (or should be) one of the principal forums for performing this function—though it is often timid in doing so.

It follows from this view of culture as a forum the induction into the culture through education, if it is to prepare the young for life as lived, should also partake of the spirit of a forum, of negotiation, of the recreating of meaning..." (p. 123)

—Jerome Bruner

The quotes are from Jerome Bruner's book, *Actual Minds, Possible Worlds*, and, it seems to me, speak directly to the issue of Logo as a culture wherein we talk about a pedagogy which includes "the spirit of a forum, of negotiation, of the recreating of meaning," (p. 123) Logo as a culture deals with how we come "to endow experience with meaning," (p. 12) The computer is a tool for the expression and negotiation of the meanings which learners ascribe to their experiences. The mode of thought required to undertake these tasks is called narrative by Bruner.

But Logo is also a computer language and as such requires another mode of thought; namely, the logico-scientific or paradigmatic in Bruner's terms. Here one is not concerned with "endowing experience with meaning," but rather with creating "...the set of possible worlds that can be logically generated and tested against observables...." (p. 13) The computer in this instance is a tool for testing the models we build.

Bruner had this to say about these two modes of thought:

There are two modes of cognitive functioning, two modes of thought, each providing distinctive ways of ordering experience, of constructing reality. The two (though complementary) are irreducible to one another. Efforts to reduce one mode to the other or to ignore one at the expense of the other inevitably fail to capture the rich diversity of thought. (p. 12)

### But What About Logo and Mathematics?

By now you may well be asking what all this has to do with Logo and mathematics. My contention is that the combined 'press' of the paradigmatic aspect of Logo and of mathematics has lead to a situation particularly in the secondary school where that 'rich diversity of thought' Bruner speaks of has been lost. I would further argue that our task as teachers of mathematics and of Logo is to foster the 'narrative' mode of thought, though certainly not to the exclusion of the paradigmatic.

But why do this, you may ask, since the essence of mathematics, if not Logo the computer language, is the power of the logico-mathematical nature of the thought characteristic of both? I think there are two reasons (at least) for encouraging the use of the narrative mode of thought in our mathematics and Logo classes.

The first has to do with mathematics in the making, or Logo programming in its active sense. This is the hypotheses generating phase, the guessing but not testing phase, the intuitive leap phase. Here one's thinking is not strictly paradigmatic, but is much more narrative in nature. "To the degree that modern science (or science in any era) also is involved in hypothesis generating, as well as hypothesis testing, it is akin to the activities of the humanist and the artist." (p. 52) Hypotheses, theorems, ways to solve problems, non-trivial programming tasks, are not items which are by and large generated in a logical, deductive fashion. It is much more the case that we use heuristics and imaginative thought in creating our problem solutions in the areas of mathematics and Logo programming.

The history of mathematics and of science is rich with stories of the narrative nature of the work mathematicians and scientists did in generating their ideas. Lakatos' book, *Proofs and Refutations*, in which he describes the growth of mathematical ideas and hypotheses, has become a modern day classic in the field of mathematics education. The accounts of Bohr and Heisenberg describing the tortured paths they took in creating their ideas are further examples of the work of the humanist and the artist in scientific and mathematical endeavors. Our students should be aware of this side of the mathematician's and programmer's life. Too often, especially in mathematics classes, students only see the end product of the work which some mathematician has done, and never get to glimpse the dead ends, the frustrations, the not so imaginative leaps which characterized mathematics in the making. This is close to what Bruner calls the narrative way of knowing, and should, I contend, play a more central role in the teaching of mathematics.

Fundamentally, mathematics itself is a paradigmatic science. The teaching of mathematics is not. Indeed, neither is education. It is a narrative enterprise, and hence all teaching is really a "negotiating and re-negotiating" of the meanings students attach to their experiences. And this is where the Logo culture can be of service to the teaching and learning of mathematics, because the cultural aspects of Logo, as distinct from the programming aspects of Logo, champion the idea of teaching being carried out in "...the spirit of a forum, of negotiation, of the recreating of meaning...." Here we can let our experiences in teaching Logo guide how we can approach with a fresh, enlivening spirit, the teaching of mathematics. Why can we not have students keep journals in mathematics as we do when teaching Logo? Why cannot the dictum of 'ask three before you ask me' apply in mathematics equally as in Logo? Why can't the mathematics classroom be organized into groups for project work as is done for Logo projects? Is it out of a fear that the students will not learn mathematics? Perhaps, but if so, then I would argue that for most students working in isolation on their mathematics classrooms has not taught them mathematics either, so what have we to lose?

And we have much to gain, I should think. If the mathematics is taught in a transactional environment, in a climate where discussion and debate, guessing and testing, trial and error, conjecture and refutation are encouraged and applauded, then the mathematical activity will acquire meaning for the students for they will have generated that meaning in their interactions with others.

### Dialog and Discussion — What Role?

What I am arguing here is not a rehash of the ideas that children must discover and invent their world all on their own. I am suggesting that dialogue and discussion, the clarification through language of thoughts and ideas, is as central to the teaching of mathematics as it is all other subjects. Listen once again to Bruner on this topic:

Some years ago I wrote some very insistent articles about the importance of discovery learning—learning on one's own, or as Piaget put it later (and I think better), learning by inventing. My model of the child in those days was very much in the tradition of the solo child mastering the world by representing it to himself in his own terms. In the intervening years I have come increasingly to recognize that most learning in most settings is a communal activity, a sharing of the culture. It is not just that the child must make his knowledge his own, but that he must make it his

own in a community of those who share his sense of belonging to a culture. It is this that leads me to emphasize not only discovery and invention but the importance of negotiating and sharing—in a word, of joint culture creating as an object of schooling and as an appropriate step en route to becoming a member of the adult society in which one lives out one's life. (p. 127)

Logo and mathematics education? What can each contribute to the other? I am suggesting here that the Logo culture, that climate in which we all wish Logo to be introduced to learners of all ages, the Samba school atmosphere if you will, is of vital importance for the teaching and learning of mathematics, perhaps even more so than the paradigmatic aspects of Logo and of mathematics. If we could have more mathematics classrooms display Logo-like environments, then I for one would be willing to bet that many more children would leave school with positive feelings about mathematics than is currently the case.

If you have experience with creating a Samba school atmosphere in your mathematics classroom, then please drop me a note describing what you have done, and I will be most pleased to report on it here.

### References:

- Jerome Bruner (1986) *Actual minds, possible worlds*  
Cambridge: Harvard University Press  
Imre Lakatos (1976) *Proofs and refutations* New York:  
Cambridge University Press

\* \* \* \* \*

An electronic message received from Ihor Charischak today (April 16th, 1988—yes, columns really are written that far in advance of their appearance) informed me that the cost of CLIME membership is now \$10.00. You can join the other 235 members of CLIME by writing to:

Ihor Charischak, 10 Bogert Avenue  
White Plains, New York, NY 10606

Membership in CLIME as well as membership in the ICCE-SIG for Logo-Using Educators will certainly keep you informed of the activities of educators who think about and experiment with the connections between mathematics and Logo.

[A. J. (Sandy) Dawson is a member of the Faculty of Education at Simon Fraser University in Vancouver, Canada. His Compuserve number is 76475,1315. He can also be reached electronically through Bitnet as UserDaws@SFU.BITNET]

## Assessing Logo Learning in Classrooms

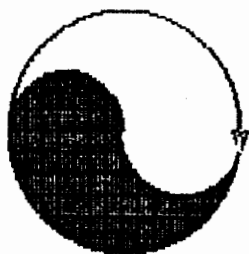
### What Are Your Logo Students Learning?

by Dan Watt

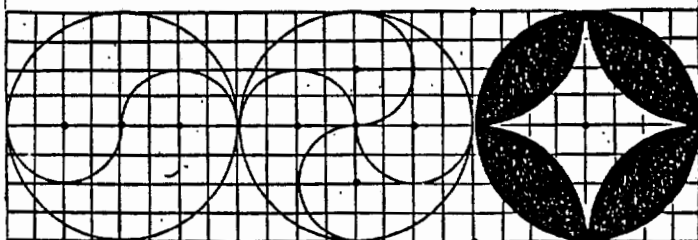
What do students learn when they learn Logo? This is the starting point for an investigation which we will be sharing together in a series of monthly columns on Assessing Logo Learning in Classrooms. Logo was designed as an integrated learning environment in which students could learn important ideas in mathematics, computer science and problem-solving, while inventing and carrying out their own computer-based programming projects. Think about your own students for a moment. What do they learn when they engage in Logo activities? Which mathematical ideas are they learning? Which ideas from computer science are they using to carry out their programming projects more effectively? And are they learning and using problem-solving strategies that can be applied to problems in other areas? This column is a partial report of some work that Molly Watt and I have been carrying out to support classroom teachers in documenting Logo learning in their own classrooms.

### Is Something Missing?

Jimmy, a sixth grade student, produced the design below, in response to a teacher challenge. His version of Logo lacked a built-in FILL command, so he had to invent his own way of filling in circles. The drawing he created was interesting enough to be chosen for the cover of the Logo Exchange (April 1987).



Teacher Challenge



Jimmy's Result

```
TO A
FULLSCREEN
HT
LT 90
REPEAT 180 [FD 57.5 BK 57.5 RT 1]
PC 0
REPEAT 180 [FD 57.5 BK 57.5 RT 1]
PU
HOME
LT 90
FD 28.75
PD
REPEAT 180 [FD 28.75 BK 28.75 RT 1]
PU
FD 57.5
PD
PC 1
REPEAT 180 [FD 28.75 BK 28.75 RT 1]
ST
PU
FD 57.5
FD 28.75
RT 90
PD
REPEAT 360 [FD 1 RT 1]
END
```

It's clear from Jimmy's design and his procedure that he had learned a great deal about Logo graphics. He used the idea that a "total turtle trip" is 360 degrees and that a turtle rotation of 180 degrees can be used to fill a semicircle. And he figured out how to create "negative fills" by drawing over a previously filled region in the background color. Furthermore, Jimmy must have done some careful experimenting to determine that the radius of a circle made by REPEAT 360 [FORWARD 1 RIGHT 1] is 57.5 turtle steps, and applied mathematical reasoning to determine that the radii of the smaller semi-circles are half that value.

But something is missing for me in Jimmy's work! There are important Logo ideas that Jimmy might have used, but didn't. For one thing, he did not give his procedure a meaningful name that would help him to remember, or allow another reader to understand, what the procedure was supposed to accomplish. For another, he used the statement REPEAT 180 [FD some distance BK the same distance RT 1], four times, without using variables, or creating a subprocedure. And then there was his use of the HOME command. This made it easy for Jimmy to move the turtle back to the place it had started, but at the same time, made it impossible to place the design anywhere else on the screen, or to draw it in a different position.

Next steps for Jimmy might include helping him learn to use subprocedures with variables, and to name his procedures effectively. Perhaps the most important thing for Jimmy to understand is how to write procedures that other people can read easily, so they can enjoy the elegant designs he creates and use his ideas as models for their own projects. And if Jimmy were to revise his procedure to make it more readable, he might also find ways to eliminate unnecessary steps, and speed up the drawing.

### Why Assess Logo Learning?

There were three major rationales for our work with classroom teachers to assess Logo learning. Jimmy's example illustrates the first: the need for classroom teachers to understand what their own students are learning, what Logo knowledge they are actually using (and not using) in their work, on a day-to-day, week-to-week basis, so that they can support them in using Logo as effectively as possible. The second rationale was to find ways that school districts which have been using Logo for some time, can obtain specific information about what their own students are learning. The third was to determine whether classroom-based research can be a path to professional development for Logo teachers. (See the references at the end of the column.)

The first reason is extremely compelling to us. Most Logo teachers want to use Logo in the way it was intended — as an environment for open-ended, mathematical problem-solving and student-initiated projects. But in order to teach effectively when students are working on their own projects, a teacher needs to know what important skills and ideas a student might learn in working through a particular activity, which of those ideas a student is actually using in her work at a particular time, and which of the ones she is not using might be especially appropriate for her to learn next, and some (individualized) teaching strategies for helping the student learn those ideas.

Recall the excitement of a Logo classroom where students have discovered how to create fascinating geometric patterns that wrap around the screen. But students often get stuck in a pattern of retyping variations of REPEAT 9999 [FORWARD 1234 RIGHT 5678] without particularly noticing the effects of changing the input numbers. Or think of a classroom in which students are working on complex projects. How often do you see a potentially lovely turtle drawing of a jet plane, with one of its wings upside-down, and with a procedure that consists of one long sequence of FORWARDs, BACKs, RIGHTs, LEFTs, PENUPs and PENDOWNs, with no subprocedures? These activities are

appropriate as one phase of a student's Logo learning experiences — especially if students use these experiences as starting points for further learning.

But if I see a classroom in which all the student programming is in "spaghetti" style, or all the students are creating elaborate wrapping designs without noticing the effects of the particular numbers they are typing, I am apt to suspect that the teacher has missed too many "teachable moments," either by not knowing what important ideas her students could be learning, or not by not having a repertoire of appropriate challenges and interventions, or — as happens far too often — by focusing only on the pictures that appear on the computer monitor, and not paying enough attention to what the students have been doing to make the turtle draw those pictures.

### How Can We Justify the Continued Use of Logo in Classrooms?

The second reason for assessing Logo learning is equally compelling. As a Logo advocate and enthusiast, I am frequently asked by teachers and computer coordinators to help them justify their use of Logo to others in their communities. "Is there any research that proves that Logo students are learning important mathematical ideas, transferable problem-solving strategies, critical thinking skills (and so forth)?" is a question I am frequently asked. "I need this information to convince my principal, department head, superintendent, or school board that our Logo program should be continued." My response is to ask them, "What are your students learning? What benefits does Logo have for them, and how do you know?"

Unfortunately, most of the answers I get are broad generalizations. "My students are learning to love mathematics." "They are solving all kinds of problems that I can't solve myself." "One of the best things for my students is that they are learning to work together and respect each other." "Some of my 'low ability' students get more involved with Logo than with anything else. It's so good for their self-confidence." Or sometimes teachers say, "Well, to tell you the truth, I haven't looked at their Logo work very much lately. I've got so many other subjects to teach. But I know it's valuable for them."

I believe that the positive responses many teachers give when asked about Logo learning are based on very real achievements of their students. And I think that hard observational data can probably be collected to support their generalizations. Nevertheless, I don't expect principals,



## Assessing Logo Learning -- continued

school boards, or superintendents to be convinced without seeing data showing some specific learning benefits for students in real classrooms in their school districts.

My own particular bias is that these questions will never be answered satisfactorily by outside researchers. And I believe that the best source of this data is classroom teachers, looking at and analyzing the work of their own students. Research conducted in other circumstances, whether the results are positive or negative, tells me little about what the students are learning in my own classroom.

### Critical Aspects of Logo Learning

These were some of the considerations which led to an 18-month exploratory, collaborative classroom research project devoted to assessing Logo learning. Molly Watt and I co-directed the project, which was funded by the National Science Foundation.

In this series of nine columns I will share our approach to understanding and assessing what we call "Critical Aspects of Logo Learning." Which aspects of Logo learning can and should be assessed by classroom teachers? Our project focussed on upper elementary grades, 4 - 6, and on ideas and skills involving mathematical thinking, introductory computer science, and problem-solving. To focus our investigations, we invented a phrase, "Critical Aspects of Logo learning." According to our criteria, these are behaviors, skills and understandings which:

- have been identified (by Logo practitioners and researchers) as essential to students effective use of Logo;
- are not usually learned by students without some form of teacher intervention
- are part of the normal learning opportunities for most Logo students;
- are readily accessible to observation by classroom teachers.

As an example of what we mean by all this, let's look at the use of clearly named subprocedures, which was missing from Jimmy's work shown above. Most of us would agree that if students don't eventually learn to use subprocedures well, they are missing out on one of the most powerful ideas embedded into Logo by its developers. However, lots of studies have shown (and our own classroom observations often confirm) that students don't necessarily learn to use subprocedures without some type of intervention by a teacher. Yet our experience confirms that the use of subprocedures can be learned by all but the youngest Logo learners,

and can be observed by any teacher who periodically examines printouts of student procedures, on screen or on paper.

To limit our observations further, we have grouped our critical aspects under eight headings:

- I. Procedures and subprocedures
- II. Procedures with variables
- III. Mathematics of Turtle Geometry — using 360 degrees
- IV. Mathematics of Turtle Geometry — other geometric ideas
- V. Learning through exploration
- VI. Debugging
- VII. Planning and carrying out projects
- VIII. Managing workspace and files

Under each heading are 4-8 subheadings, identifying specific behaviors, skills or concepts, arranged roughly in the order in which students typically develop them. In each of the months that follow, I will describe the specific learnings listed under one of the eight headings, illustrated by examples of student work. You will notice that our list of critical aspects is far from a complete survey of everything students can learn by using Logo. What we are attempting to do is cover the fundamentals, aspects of Logo that most teachers want and expect their students to learn in grades 4 -6, and develop a common language and common descriptions for them. You will also notice that we are not trying to document transfer of Logo learning to non-Logo domains. We believe that it is premature to attempt to observe transfer of Logo learnings until we can document what Logo students are actually learning in Logo itself.

The work I will describe was very much a group effort. Our Research Assistant was Joan Funk, and our collaborators were nine computer coordinators and classroom teachers from four Greater Boston communities: Elizabeth Berner, David Crump, Diana Freedman, Doreen Kelly, Robert Kondel, Jane Manzelli, Mary Miller-Teahan, Pasquale Puleo and Kitsy Rothermel. All of these folks contributed to the development of the ideas and materials I will be sharing here. However, the particular ways these ideas are expressed are those of Molly Watt and myself, and do not necessarily reflect the views of any of our collaborators, or of the National Science Foundation.

### A Clearinghouse for Assessing Logo Learning in Classrooms

I also plan to devote part of these columns to other ways of assessing Logo learning. If you'll write to me about what

you are doing in your classroom or school district, I'll summarize it and share it with other readers. If you have materials or articles you're willing to share, please tell me where our readers can get them and how much they should send to cover copying and mailing.

Our own work is not yet in final form and we are submitting proposals to refine it further during the next year or two. Your participation in this process could be very important to us. We'd like to know your responses to both our general approach, and to the specific items and their descriptions. I am very much looking forward to sharing this process with participants in the Logo Exchange.

### References

- Watt, M. and Watt D. (1986). *Teaching With Logo: Building Blocks for Learning*. Addison-Wesley, Menlo Park California.
- Watt, M. and Watt, D. (1987). *Using the Process Approach to Teaching Writing as a Model for Teachers of Logo*. Proceedings, Third Conference on Logo and Mathematics Education (LME3). Concordia University, Montreal Canada.
- Watt M. and Watt, D. (1988). *Teachers as Collaborative Researchers: Professional Development Through Assessing Logo Learning*, Proceedings, National Educational Computing Conference 1988. Dallas, Texas.

If you would like copies of the two papers listed above, please send \$5.00 for photocopying, postage and handling, and a pre-addressed mailing label, to Angela Daskalos, Center for Learning Technology, Education Development Center, 55 Chapel Street, Newton, MA 02160. Please specify the name(s) of the paper(s) you want.

Daniel H. Watt, Educational Alternatives,  
Gregg Lake, Road, Antrim, NH 03440

### Call for Papers Second European Logo Conference August 30 - September 1, 1989

Contact:  
State University of Gent  
Department of Education - EDIF  
EUROLOGO '89  
H. Dunantlaan 1  
B9000 Gent, Belgium

## Logo Connections

### The Journey from Lego to Logo

by Glen L. Bull and Gina L. Bull

The idea of connecting Logo with devices in the external world was embodied in the original spirit and vision of Logo. Evidence of this is found in Logo Memo Number 3, "Twenty Things to Do with a Computer," written in June 1971 by Seymour Papert and Cynthia Solomon. The memo discusses sensors which Logo can use to detect events in the outside world:

Touch sensors, light sensitive cells and sound detectors are obvious examples that require very little cleverness. Accelerometers and tilt detectors lead to more sophisticated fun. (pages 3-4)

The memo also discusses use of Logo to control external devices:

The school computer should have a large number of output ports to allow the computer to switch lights on and off, start tape recorders, actuate slide projectors and start and stop all manner of little machines." (page 27)

The remarkable aspect of this memo is that it was written at a time when very few people believed that it would be possible to have a computer in every classroom, or that children could control them. At that time, the cheapest computers cost \$100,000 and the smallest ones were the size of a refrigerator. In the 1980's the vision of computers in every school came true. The vision of computers which listen to sensors and talk to motors is now coming true as well.

### A New Theme for the Year

"Logo Connections" is the fourth title in a series of columns which we have written since the founding of the Logo Exchange. Prior columns include "Microworlds" (1982-84), "Listful Thinking" (1984-86), and "Teaching Tools" (1986-88). Each of the columns had a theme; for example, the "Listful Thinking" series was about using lists with Logo, while the "Teaching Tools" columns were about use of tool procedures in the classroom. The "Logo Connections" column will follow in this tradition.

Each of the articles in this series will be focused around the theme of connections between Logo and the world. This column will be about Lego-Logo and videodisc players, and speech synthesizers, and robotic arms, and joy sticks, and touch screens, and telecommunications.

## Logo Connections -- CONTINUED

Much of what is good about Logo thrives best in a community of Logo users. Often information needed to interface Logo to the world is not written down in a single place (although we'll attempt to include some of this information in this column). Hence, "Logo Connections" is also about connections among people who use Logo.

Special versions of Logo, such as Lego-Logo, now make it possible for the computer to do many of the things outlined in Papert's original vision. In the 1970's it seemed unlikely that computers would be available in schools throughout the country. In the 1980's classroom computers have arrived. The 1990's will see the routine use of classroom computers with sensors and motors and peripheral devices.

Surprisingly challenging problems may be embodied in relatively simple ideas. Papert suggests projects such as construction of a mechanical yoyo, or development of a robot turtle that can balance a broomstick on its nose. A semester-long curriculum could be built around either of these ideas. For example, how could the computer control the yoyo? Would a yoyo on the end of a stick rotated by a motor work? Once computer control of the yoyo is possible, what kinds of programs are needed? Would it be possible to write a program to do a trick like Waking-the Dog or Round-the-World? These projects involve many domains: physics, engineering, and problem solving. Since they are likely to be tackled as group projects, social skills and interactions are also involved.

Many scientists and engineers first began by tinkering as children. Biographies of today's scientists often include anecdotes of children taking apart and reassembling radios and watches and other devices. A child who takes apart a radio today is likely to find a block of encapsulated plastic. A young tinkerer who takes apart a watch may find a digital chip instead of a mainspring.

If advances in technology are taking away some of the traditional avenues for tinkering, technology can replace these lost opportunities with other possibilities. For example, Lego-Logo is a special version of Logo designed specifically to foster tinkering. The kit comes with gears and motors and pulleys and a special computer interface to control them. For any child who has constructed a tower with Lego blocks, the transition to combining a computer and Lego is a natural one.

Many computer languages have a high ceiling, but few have a high ceiling and a low threshold. Lego-Logo is appropriate for physics and mathematics class. Later this year we will describe a high school vocational teacher who

uses Lego-Logo in automotive shop class. However, Lego-Logo is also appropriate for younger children.

## Stephen Studies Science

Last week Stephen spotted a group of adults tinkering with a new toy, and asked to play too. Initially he was told, "This is a toy for older children." However, he quickly noticed that the toy was constructed from Lego bricks, and gave us an incredulous look as he pushed his way to the front. Wasn't this Lego? ...and isn't Lego his particular domain of expertise? It is an odd thing, but the curiosity of a child can far outlast the patience of any adult. A strength of the computer is that its patience outlasts that of any adult. For example, talking word processors such as Talking Textwriter (available from Scholastic) permit children to repeatedly type words and letters to hear how they sound. Teresa Rosegrant has conducted research that demonstrates that a young child may type a single letter more than fifty times just to hear it over and over. Most adults are likely become bored after reading a word to a child a dozen times, but the computer will say the word as often as the child wants. (Logo and Talking Textwriter are both examples of "Learner Based Tools," because they allow the child to control the computer.)

For most of an evening Stephen was fascinated by the ability to control the motor by typing ON and OFF. Lego-Logo has several new commands. ON and OFF are used to turn motors on and off. From time to time Stephen would dash upstairs to ask for help. For much of the evening he attempted to type "ON" and "OF" rather than "ON" and "OFF". There were also a number of times when "ON" and "NO" were interchanged. By the middle of the evening he had convinced himself that the computers only accept exact spellings. He soon pushed two Lego struts onto the shaft of the motor to make a fan. He happened to select a red and a blue strut, and found that purple is produced when red and blue are spun. This led to a joint investigation of various color combinations such as two red struts and a blue strut, or two blue struts and a red strut.

In the process of this investigation, Stephen accidentally selected a long strut and a short strut which produced an asymmetric blade. He became fascinated by the fact that asymmetric blades are unstable, and likely to fly apart in a spectacular fashion! We spent a considerable amount of time determining which combinations made the most noise and flew the farthest.

Later the fan metamorphosed into a propeller and became a plane. The Lego-Logo interface box has five ports which can be used to control motors and lights. The propeller

plane was connected to Ports 2 and 3. A red light for one wing was connected to Port 2 and a green light for the other wing was connected to Port 3. In Lego-Logo, the TALKTO command is used to identify which port is being addressed. The command to turn the red light on would look like this:

```
TALKTO 2
ON
```

Similarly, the command to turn on the green light connected to Port 3 would look like this:

```
TALKTO 3
ON
```

It is possible to talk to several ports at once in this way:

```
TALKTO [2 3]
ON
```

However, these commands were a little abstract for Stephen, so we defined the following commands with Logo procedures:

```
TO FAN
TALKTO 1
END

TO RED
TALKTO 2
END

TO GREEN
TALKTO 3
END
```

These procedures permitted a syntax such as:

```
FAN ON
RED ON
GREEN OFF
```

This was the beginning of a two-word Logo grammar for Stephen. Later he wanted to turn on both lights at once, so we created the following procedure:

```
TO LIGHTS
TALKTO [2 3]
END
```

A strut on the shaft of a motor is a relatively simple Logo world compared with more sophisticated projects which are possible with Lego-Logo. However, even simple projects such as this lead to a wide array of investigations. The fascination that comes from control of a three-dimensional fan motor is difficult to describe, but it permits explorations that are not possible with the two-dimensional screen of a

computer.

Throughout the best part of a week Stephen explored the physics of light, and angular momentum, and digital control systems. (He thought he was playing with a propeller.) Even the best CAI tutorial or computer simulation on a two-dimensional screen could not have duplicated the kinds of exploration that were possible with the actual objects. In Stephen's case, the benefit that we perceive is not specific to a particular content that might be learned. Rather, he is using Lego-Logo to learn about the process of tinkering.

Two decades ago Papert described his vision of a school of the future:

In our image of a school computation laboratory, an important role is played by numerous 'controller ports' which allow any student to plug any device into the computer. The ports are protected by fuses and suitable interfaces so that little harm will be done if anyone carelessly puts the main voltage into a computer output port. The laboratory will have a supply of motors, solenoids, relays, sense devices of various kinds, etc., etc. Using them the students will be able to invent and build an endless variety of cybernetic systems." (page 39)

That early portrait painted by Papert is a fairly good description of what can be accomplished with Lego-Logo.

#### Reference

Papert, S., & Solomon, C. (1971). *Twenty things to do with a computer*. Artificial Intelligence Memo No. 248, Logo Memo No. 3, Cambridge, MA: Massachusetts Institute of Technology.

*Glen Bull is a professor in the University of Virginia's Curry School of Education. His BitNet address is GLB2B@VIRGINIA. Gina Bull is a programmer analyst for the University of Virginia Department of Computer Science. By day she works in a Unix environment; by night in a Logo environment. Her BitNet address is RLBOP@VIRGINIA.*

# Variables and Problem Solving

by Dave Moursund and Sharon Yoder

## Generalizability

Generalizability is an important idea in problem solving. Generalizability means that one works to understand and solve a whole category of problems, rather than just a specific instance of a type of problem. For example, suppose that we want to draw a square with a word inside. If we need only one particular size of square and one particular word, we could print stickers or labels that contained the solution. People needing squares with the word inside could carry around the stickers.

But squares come in all sizes, and there are lots of different words. Thus, we are more likely to want a general technique for drawing a square of any size, and a general technique for adding any specified word. With these two general techniques in hand, we can solve every problem that involves drawing a (variable-sized) square and inserting a (variable) word.



Generalizability lies at the heart of mathematics. When presenting "age problems," math teachers really don't care how old Mary and Jack are. Rather, they are interested in their students learning general techniques, first for solving every age problem, then every word problem with characteristics similar to age problems, and finally word problems in general.

Similarly, students learning Logo soon find a real need to write increasingly general procedures. Here is a procedure to make a square of size 50.

```
TO SQUARE
REPEAT 4 [FORWARD 50 RIGHT 90]
END
```

A different procedure is needed if we wanted a square of side 70. A better approach is to write a general procedure that draws squares of all sizes. In Logo we would write a procedure that makes use of a procedure input. A procedure input is a special kind of variable.

```
TO SQUARE :SIZE
REPEAT 4 [FORWARD :SIZE RIGHT 90]
END
```

This procedure is a generalization of the square procedure that can draw a square of only one size.

## Variables

Humans make intellectual progress by building on the previous work of others (See May, 1988 LX). Often it takes hundreds of years for humans to fully develop an idea, give it a precise definition, and infuse it into general use. Initially a new idea may be a topic of discussion for only the most educated parts of the population. But eventually it may filter down to the general population, and perhaps even to grade school students.

The idea of variable has these characteristics. It is one of the most important ideas in mathematics. It is a powerful idea, useful throughout all of mathematics and its applications. But it is a difficult idea. No wonder that research indicates students have a hard time learning to understand and use the concept of variable. (For details, see Doug Clements column "Testitudinal Testimony," LX, January - April, 1988.)

It is often suggested that students who encounter the idea of variable in computer programming will gain a useful, working knowledge that they can apply in areas outside of computer programming, such as in algebra or mathematical applications. Perhaps students who learn to make use of variables in Logo early in their educational careers will gain a distinct educational advantage over students who do not learn to use Logo.

Some research suggests that this is the case. If a Logo teacher has a good understanding of the concept of variable and provides appropriate learning opportunities, students can gain a good and useful understanding of some of the important ideas and applications of the idea of variable. This can be quite helpful in a broad range of problem-solving situations.

Let's turn next to a couple of examples of variables.

## First Example: Pig Latin

Consider the following set of rules for converting words into Pig Latin.

1. If a word begins with a vowel (a, e, i, o, u) add the letters ay at the end. Some examples:



| <u>English Word</u> | <u>Pig Latin Word</u> |
|---------------------|-----------------------|
| about               | aboutay               |
| eat                 | eatay                 |
| into                | intoay                |

2. If a word does not begin with a vowel, take all of the consonants up to the first vowel, move them to the end of the word, and add ay. Some examples:

| <u>English Word</u> | <u>Pig Latin Word</u> |
|---------------------|-----------------------|
| cat                 | atcay                 |
| through             | oughthray             |
| seal                | ealsay                |

Now suppose that I want to write a Logo procedure that will ask the computer user to type a word and will then display the word and its Pig Latin equivalent. When we talk about this exercise we might use sentences such as "The word that the computer user is going to type." and "The Pig Latin equivalent that the computer is going to display on its screen." Each of these sentences is a *generalized* way to represent the many possible specific values. We can say that each of these sentences is the *name of a variable*.

#### Name of the variable

The word that the computer user is going to type

#### Some possible values of the variable.

cat  
house  
George  
enough

#### Name of the variable

The Pig Latin equivalent that the computer is going to display on its screen.

#### Some possible values of the variable

atcay  
ousehay  
eorgeGay  
enoughay

Notice that the variable name "The word that the computer user is going to type." may have have associated with it quite a few different values. Indeed, there are about a half million different English words and there are thousands of proper names. Notice also that it is awkward to continually use the long sentence "The word that the computer user is going to type." If one is going to be talking about this quite a bit, it might be more convenient to adopt some sort of shorthand notation for it.

Each programming language has its own restrictions for shorthand notation for naming variables. In Logo, for example, we might use `:INPUT.WORD` or `:ENGLISH.WORD` or perhaps just `:ENGLISH` for the shorthand notation. In a similar manner, the variable name "The Pig Latin equivalent that the computer is going to display on its screen" might be represented in Logo by `:PIG.LATIN.WORD` or `:PIG.LATIN`.

## Second Example: Summing the Digits

Consider the problem of writing a procedure that allows the computer user to type a number that is a positive integer and has the computer display the sum of its digits. Once again we have two variables which we need to name so that we can talk about them. We might name them "A positive integer to be typed by the computer user." and "The sum of the digits in the positive integer.." Examples include:

#### Name of the variable

A positive integer to be typed by the computer user.

#### Some possible values of the variable

4371  
527  
29435

#### Name of the variable

The sum of the digits in the positive integer.

#### Some possible values of the variable

15  
14  
23

Once again we see the advantage of using more concise notation if we are going to write a computer program. We need a way to represent the values that are input and the values produced in a general way. In Logo we might use the variable name `:POSITIVE.INTEGER` for the first variable and `:DIGITS.SUM` for the second variable.

In the above discussion we have talked about variable, name of a variable, and value of a variable. The examples suggest meanings of the latter two terms. Mathematicians define variable as follows: (James and James)

[A variable is] a symbol used to represent an unspecified member of some set. A variable is a "placeholder" or a "blank" for the name of some member of the set. Any member of the set is a value of the variable ...

## Variables and Problem solving-- CONTINUED

This definition implies that there is no difference between the term variable and what we have called the name of a variable. Let's next explore further the meaning of value of a variable.

### Values of a Variable

In each of the above examples of a variable there is a set of allowable values for the variable. In the first example, the variable represented by :ENGLISH.WORD is any of the 500,000 or so words in the English language and any proper names spelled in English. If we wrote a procedure to convert English to Pig Latin, then we would not expect it to work for groups of characters such as ?<\*&?%#%. In the second example, the variable :POSITIVE INTEGER can take on values such as 142 and 93324, but a procedure to add the digits of an input number would likely not work on values such as 2.45, 4 1/2, or -32.

More generally, each variable has set from which its values must come, just as each problem has a set from which possible solutions to that problem must come. An equation to be solved in algebra is certainly not going to have a solution found in a collection of photographs. Similarly a problem involving the shortest route between two cities is not going to have its solution among a collection of house plants. These examples seem ridiculous because we come to the solution of problems with an almost subconscious awareness that the solution must lie in some particular collection of values. This is a very important idea about variables.

Even very beginning Logo users quickly encounter the idea of a legal set of values which various Logo primitives can process. The primitive FORWARD expects an input (a specific value of a variable) from a limited subset of the real number system. If you type

```
FORWARD HELLO
```

Logo responds with a message like

```
I DON'T KNOW HOW TO HELLO
```

HELLO is not in the set of legal values that FORWARD can use.

Using the procedure SQUARE given earlier in this article, we can explore the idea of appropriate values for its procedure input :SIZE with such experiments as

```
SQUARE 50
SQUARE 23.6
SQUARE 1 / 5
SQUARE -234
SQUARE 99999999
```

Except for the last example, all of the above will draw squares in most versions of Logo. This last example is dependent on the size of number allowed by the version of Logo you are using. But what happens if you try other types of Logo objects as input?

```
SQUARE "HELLO
```

gives

```
FORWARD DOESN'T LIKE HELLO AS INPUT IN
SQUARE
```

```
SQUARE [HELLO]
```

gives

```
FORWARD DOESN'T LIKE [HELLO] AS INPUT
IN SQUARE
```

```
SQUARE "FIFTY
```

gives

```
FORWARD DOESN'T LIKE FIFTY AS INPUT IN
SQUARE
```

While the SQUARE procedure accepts any Logo object as input, when that value is given to FORWARD, FORWARD objects and tells you that "HELLO, [HELLO] and "FIFTY are not in the allowable set of values that can be processed by FORWARD. To use Logo terminology, we say that "HELLO, [HELLO] and "FIFTY are not legal inputs to FORWARD. The last example is particularly interesting because the intent is quite clear to a person reading the line, but Logo is unable to make the translation between the word for a number and the number itself in the manner that a human being would.

### Transfer Examples

There are three key ideas discussed in this article. The first is generalizability—working to understand and solve a whole category of problems rather than specific instances of problems. This is an absolutely essential idea in education. Teachers must help students to learn this fundamental idea and apply it in all of their learning. The second idea, that of variable, is inextricably intertwined with that of generalizability. As we talk about a general class of problems, and general methods for attacking these problems, we are naturally led to use of variables.

The third idea is that of the domain or set of values from which "legal" values of a particular variable must come. We will address this idea in more detail in the next article in this series.

The transfer examples given below suggest some ways you can help your students to better understand generalizability and variables.

1. Consider classification in science. For example, the word "bird" is used to describe a set. It is a shorthand for "A member of the class Aves all differing from the ancestral reptiles in possession of a covering of feathers instead of scales..." Robins and seagulls are specific examples. A conifer is a type of tree, and specific examples include douglas fir, blue spruce etc.

2. Consider the following paraphrased statement from logic:

All people are mortal.  
Pat is a person.  
Therefore, Pat is mortal.

Logical statements like this often contain the name of a variable and an instance of that variable. Here, Pat is a specific instance of "people."

3. An obvious example comes from the math classroom. Whether you are considering problems such as

$$\square + 5 = 12$$

or problems of the form

$$x + 5 = 12$$

you are working with the concept of a variable. The set from which solutions to these types of problems come depends on the age and sophistication of the students.

4. Consider a discussion in a social studies class. The discussion might be about civil wars. Students could be asked to give specific examples. "Civil war" is the name of a variable. Not all wars that the students have studied would be in the set of all civil wars, since not all wars are civil wars.
5. Music is divided into many categories: baroque, classical, jazz, rock, new age, and so forth. Often students are only aware of particular instances of music, and not the general categories. You could work from the specific to the general and discover that there are indeed "variable names" for music. Someone quite knowledgeable about music could come in as a guest speaker to describe the characteristics that put music into these categories.

6. We use general categories in our classrooms all of the time. We refer to "Tuesday's math assignment" or "last week's spelling list." These descriptions are indeed variable names. You might challenge the class to find as many of the general categories as they can that are used during a particular day.
7. We also use categorizations to manage classrooms or even schools. We talk about "the first row" or "everyone with last name beginning with C" or "all of the seniors." How many of these variable names do your students understand? How many of them are obvious? How many did they have to learn when they first entered your school?

Keep in mind as you look for ways to transfer the ideas in this article to other areas of the curriculum that you want to emphasize the idea of generalizability. Be sure to get beyond just thinking about naming, specific values, and solution sets. The powerful idea here is to learn to generalize your strategies to a wide variety of problems.

#### Reference:

James, Robert (1968) *Mathematics Dictionary*, D. Van Nostrand Company, Inc.: Princeton, New Jersey.

Dave Moursund and Sharon Yoder, ICCE, 1787 Agate Street, Eugene, Oregon, 97403; Ph: 503-686-4414; CIS: 73007,1645; BitNet: ICCE@Oregon

Now Available

### Introduction to Logo Programming Using LogoWriter

by  
Sharon Burrowes Yoder

For more information, contact:

ICCE  
University of Oregon  
1787 Agate Street  
Eugene, Oregon 97403  
503-686-4414

## Search and Research

### Early Experiences with Mathematics

by Douglas H. Clements

The research on Logo and the learning of mathematics keeps coming in, but we really must move on to other areas! Therefore, this month's column will wrap up the topic—temporarily—by discussing several investigations of children's early experiences with Logo-based mathematics. Future columns will review research in such domains as problem-solving, sex differences, special education, students' programming strategies, and effective teaching techniques.

#### Number and Shape

Hughes and Macleod (1986) claim that children have a good concrete grasp of number, but need to be introduced to written numerals in such a way that they can see clearly what function the numerals are serving. For this reason, the researchers used Logo as a transitional object between the concrete world and the abstract world of mathematics. They worked with 15 six-year-old children from a severely deprived area in London, England. These children had extremely limited knowledge of computers when they began the project. Most could recognize and name basic shapes, but few could give adequate directions on how to draw them. Their competence with numbers up to 10 was adequate, but below average for their age. The researchers introduced a simplified Logo to the children, including both a floor and screen turtle and a simplified keyboard arrangement.

The researchers claim that their project was a "considerable success." The children made gains on all subscales of a standard abilities test, although only three were significant: block design, digit recall, and basic number. Thus, there seemed to significant effect on number and shape, but not other aspects of their development. Interviews with the staff confirmed that children showed increased mathematical understanding, especially in the areas of number and shape, and particularly with regard to increased awareness of the properties of shapes and the meaning of angles. Children also engaged in high levels of mathematical discussion. For example, three boys—Kevin, John, and Andrew—came up with the idea of drawing a snowman (their completed project is pictured in Fig. 1).

They begin with a hexagon. Jon asks if it should be the tummy or the head. John says, "That can be the body. You could easily make that be the head if you want, but it'll go off the paper" (p. 197). The boys agree, and also agree that the pen should be raised before moving to the top of the body to

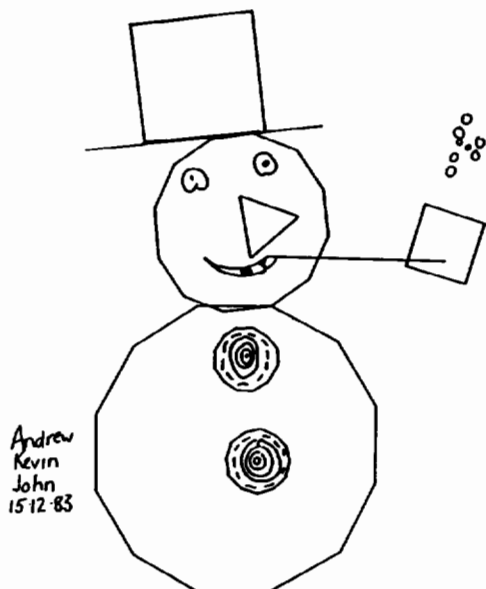


Figure 1

make the head. At the keyboard, Kevin begins to turn to the left, but Andrew corrects him, and together they place the turtle into position (citations from pp. 197-198).

John: How big do you want it to be? One or two? ...Dinnae press "nine".

Andrew: Three.

Kevin: Then "GO"

The turtle draws the head.

John: That's quite a good snowman.

Kevin: Buttons.

Andrew: It'll be easy to make buttons, you just press "one" and "circle".

Kevin: Hexagon, hexagon.

Andrew takes over at the keyboard and moves into the middle of the snowman's body.

Kevin: Make buttons now.

Cathie: What size are you going to make them?  
[the teacher]

Kevin: Small.

Andrew: Mmmmm...one.

John: I say about one.

Kevin: It's very good!

More buttons are made and the boys discuss what shape to draw for the hat. Each has an idea: triangle, circle, or square. John takes control of the keyboard and they decide

on a square so that a brim is possible. They wonder, again, what size to use.

Kevin: Small.

John: Two.

Andrew: Shh, listen to what Cathie says. What number?

Cathie: I don't know. John, I think you should decide, it's your hat.

Kevin: Should take four.

Cathie: I think that would be too big, wouldn't it?

Kevin: You've never took four before.

Andrew: Do three...It'll be too big.

John: I say two.

Cathie: If you say two, put in two, John. Put in two. [The square is drawn]. That's fine, isn't it?

They then decide a length for the brim.

Andrew: Two...One....

John: Try one.

Kevin: And then do another one?

Cathie: Well, it's got to go to the corner, and one past....

John: Right, two....Will I go forward?

They decide on a triangle for the nose, and a "square on the end of a line" for a pipe. In this and similar projects, children frequently discussed the relative size of drawings produced by various number inputs, appropriate shapes to use, and so on. The teachers found it striking that these children became able to estimate whether to send the turtle a small distance or a longer distance, for instance, whether a line had to be a four or a five.

Similarly, Bowman (1985) reported that Logo experience heightened 4- and 5-year-old children's interest in numerals, and they learned them more readily than in previous years. These children, too, learned the relationship between the length of a number line and the size of the number entered. Such benefits may take time to evidence themselves, however. Akdag (1985) also worked with kindergartners, for a relatively short period of time. She found no treatment effects on mathematics readiness, Piagetian, and Map tests, although she did report gains by the Logo group on a geometry concept test.

With time and guidance, however, Logo may help promote an even wider range of numerical skills. Hines (1983) worked with five children for 10 weeks on preskill games (including orientation to the computer and body orientation activities) and Logo drawing. The children increased in their ability to identify the larger of two numbers and conserve number. Of course, the small number of children involved and the lack of a control group limit

generalizability of these findings. Nevertheless, the ability to visualize the differences in number inputs in Logo may have helped children concretize number quantity.

As another example of this "concretization," a second grader wrote a program called COUNTDOWN that contained a "bug." Instead of counting to zero and printing "BLAST OFF!" it printed, "...4 3 2 1 0 -1 -2 -3 -4...." There was something wrong, but the bug was marvelous to the boy. He had heard something about negative numbers, but had never seen any generated in a natural and believable way. He fixed the bug, but may have learned even more from the "mistake" (Feurzeig & Lukas, 1971).

## Arithmetic

Logo does not provide efficient practice on arithmetic processes. However, it does provide a context in which there is a real need for these processes and in which children must clearly conceptualize which operation should be applied. For example, first grade children determined the correct length for the bottom line of their CLOCKTOWER (Fig. 2) by adding the lengths of the three horizontal lines that they constructed at the top of the tower:  $20 + 30 + 20 = 70$  (Clements, 1983-84).

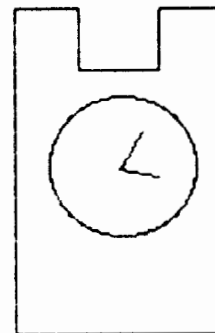


Figure 2

A primary grade teacher of a "below average" class reported that she had written some multiplication tables on the board. The students saw the pattern immediately, using the word REPEAT to describe what they noticed. She claimed that "This is the first time this has ever happened in my class. I am sure it is because of their Logo work. They are looking for patterns in things more than they used to do" (Carmichael, Burnett, Higginson, Moore, Pollard, 1985, p. 286). Many students developed a new sense of arithmetic, numbers, and estimation within the context of their Logo

## Search and Research -- CONTINUED

work. Thus, even in the domain of arithmetic, work with Logo can give students the opportunity to view mathematics in new ways.

### Measurement and Estimation

Previously, we reported that Kull (1986) observed first grade children using standard units of measure such as 44 or 66 (they said that they "wanted a big number and it's easiest to hit the same one twice") to make their drawings, such as a rectangle with a width of 44 and a length of 88 (FD 44 FD 44). Campbell (1987) also believes that Logo can help young children learn about measurement (as well as help us learn more about what young children learn about measurement). Why? She claims that Logo provides an arena in which young children may use units of varying size, define and create their own units, maintain or predict unit size, and create length rather than end point representations through either iterative or numeric distance commands. Further, Logo permits the child to manipulate units and to explore transformations of unit size and number of units without the distracting dexterity demands associated with measuring instruments and physical quantity.

Working with kindergartners and first graders, she found that:

- Children have difficulty adjusting to changes in unit size, especially when the unit size is halved. They may be using perceptual/spatial strategies rather than numeric/mathematical strategies to solve these problems.
- Children are, however, remarkably accurate when estimating halved distances, even if the mid-point of the total distance has to be imagined.
- Children understand that a distance traversed with a smaller unit requires a greater number of units than that same distance traversed with a larger unit. However, they consistently underestimate the strength of the inverse relationship between unit size and numeracy of units.
- Contrary to expectations, estimation of lines with oblique orientations is not usually more difficult than lines with horizontal or vertical orientations. There was no differences across grades; a rather modest amount of Logo experience (2 hours) may have helped the kindergarten children ignore the problem-irrelevant variables in the spatial field.
- In almost all comparisons, Logo children were more accurate than control children. The control children were more likely to: underestimate distances, particularly the longest distances; have difficulty compensating for the halved unit size; and underestimate the inverse relationship between unit size and unit numeracy. Thus, in

contrast to the Logo children, the control children showed no evidence of a "big" number strategy. The Logo experience may have contributed to estimation accuracy and independence from the distracting features in the spatial field. Nevertheless, even inexperienced children knew that the smaller numbers were to be associated with the shorter lengths and that progressively larger numbers would be assigned to progressively longer lengths. This basic principle of measurement seems to be acquired early in life and may not be dependent on specific measurement experiences.

### Conclusions

These studies indicate that if Logo is used thoughtfully, it can provide an additional evocative context for the young children's explorations of mathematical ideas. Teachers should never forgo concrete experiences, of course. Logo experiences are supplements and extensions, not replacements. Nevertheless, the (recently popular) opinion that children must reach a certain age or stage (e.g., concrete operations) before successfully using Logo appears invalid. Wise use of appropriate Logo environments can enrich early mathematical experiences.

### References

- Akdag, F. S. (1985). *The effects of computer programming on young children's learning*. Unpublished doctoral dissertation, Ohio State University.
- Bowman (1985, November). *Computers and young children*. Paper presented at the annual meeting of the National Association for the Education of Young Children, New Orleans.
- Campbell, P. F. (1987). *Measuring distance: Children use of number and unit*. Final report submitted to the National Institute of Mental Health Under the ADAMHA Small Grant Award Program. Grant No. MSMA 1 R03 MH423435-01. University of Maryland, College Park.
- 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.
- Clements, D. H. (1983-84, December/January). Supporting young children's Logo programming. *The Computing Teacher*, 11(5), 24-30.
- Feurzeig, W., & Lukas, G. (1971). LOGO—A programming language for teaching mathematics. *Educational Technology*, 12, 39-46.
- Hines, S. N. (1983, July-August). Computer programming abilities of five-year-old children. *Educational Computer*, pp. 10-12.
- Hughes, M., & Macleod, H. (1986). Part II: Using Logo with very young children. In R. Lawler, B. du Boulay, M. Hughes, & H. Macleod (Eds.), *Cognition and computers: Studies in learning* (pp. 179-219). Chichester, England: Ellis Horwood Limited.
- Kull, J. A. (1986). Learning and Logo. In P. F. Campbell & G. G. Fein (Eds.), *Young children and microcomputers* (pp. 103-128). Englewood Cliffs, NJ: Prentice-Hall.

Douglas H. Clements, SUNY at Buffalo, Department of Learning and Instruction, 593 Baldy Hall, Buffalo, New York 14260 ;  
CIS: 76136,2027, BuNet: INSDH@UBVMSA



## Global News

**Edited by Dennis Harper**

University of the Virgin Islands  
St. Thomas, USVI 00802

| AFRICA                 | ASIA               | AUSTRALIA              | EUROPE                 | LATIN AMERICA        |
|------------------------|--------------------|------------------------|------------------------|----------------------|
| Fatimata Seye Sylla    | Hillel Weintraub   | Anne McDougall         | Harry Pinxteren        | Jose Armando Valente |
| Lab Informatique et Ed | Doshisha Int. Sch. | Faculty of Education   | Logo Centrum Nederland | NIED                 |
| BP 5036 Dakar          | Tatara, Tanabe-Cho | Monash University      | P.O. Box 1408          | UNICAMP              |
| Senegal, West Africa   | Kyoto-fu Japan     | Clayton, Victoria 3168 | BK Nijmegen 6501       | 13082 - Campina      |
|                        | 610-03             | Australia              | Netherlands            | Sao Paulo, Brazil    |

### News From Dallas

I am writing this column from the Anatole Hotel in Dallas where the NECC 88 Conference is now taking place. An all-day preconference symposium was held by the International Committee of the ICCE and some of the content of that meeting have some ramifications for the global Logo world. As the ICCE has grown in importance and members, perhaps it would be beneficial for our foreign readers to describe some of its branches and activities. The ICCE produces *The Computing Teacher* (TCT) as its journal (readership about 13,000). The TCT will have two new columns in the upcoming volume, one dealing with international events and the other with telecommunications. Both these columns directly address the international community. The "I" in ICCE stands for International and the group is attempting to become the major voice for computer educators throughout the world.

The ICCE has sub groups:

- (1) Organizational Members (OMs)
- (2) Special Interest Groups (SIGs)
- (3) Committees

One of the special interests groups (and by far the largest) is SIGLogo. If you subscribe to this journal then you are a member of SIGLogo. SIGLogo was officially recognized at the NECC and had its first meeting last night. Of international note at this meeting was a proposal to establish a list of Logo educators throughout the world that could be used by SIGLogo members to contact or visit while traveling abroad. These contacts and visits would help enable Logo to spread through mutual contacts. Another suggestion was to update the Logo Bibliography that Tom Lough and myself put together for the Logo 86 conference. If you are reading this from a foreign country please send any Logo books, research and article references from your country to my address at the end of this column.

Organizational members are groups of computer using educators who form organizations and do local conferences

and activities but come under the umbrella of the ICCE and receive support from them. Many countries have organizational members and those interested should contact the ICCE or myself.

### The International Committee at Work

Committees are periodically appointed by the ICCE board to conduct business, workshops, conferences and other events. One such committee was established for international computer education. Three working meetings have been held (Tel Aviv, Edinburgh, and Dallas) where international concerns have been addressed. Some of the recommendations made during the past year include:

- Back issues of magazines be sent to developing countries.
- Joint software development take place between nations.
- International software exchange be established.
- Adaptation and development of software be addressed.
- International training workshops for software developers be held (the first one is scheduled for Sweden).
- Need for the ICCE to sponsor international conference addressing a range of issues.

The all-day Dallas symposium emphasized the role of telecommunications in international education. Numerous companies described joint global projects. Other speakers gave their views of the potential benefits and pitfalls of using communications. A list of recommendations to ICCE regarding the use of telecommunications in education throughout the world was produced and will be submitted to the ICCE board.

A difficult question arose as to how developing countries could benefit from this technology. It was suggested that *The Computing Teacher* and the *Logo Exchange* be sent to USIA and teacher training colleges free-of-charge in developing countries so they would at least be aware of what was happening.

Any reader interested in the international committees work should contact me at the address below.

## Global News -- continued

### We Welcome a New Editor

The *LX* would like to welcome Jose Valente of Brazil as our new Latin American editor. Many of us know Jose from past Logo conferences and it is indeed an honor to have a man of his stature submitting Logo articles to this column. Next month, this column will again have reports from our continental editors.

My first column as Latin American editor is about the Logo activities in Brazil. Initially I would like to give an overview about the use of computers in education in general in this country, and then concentrate on Logo.

### Computers in Education in Brazil

The use of computers in education in Brazil — including Logo — is mainly sponsored by government, specifically the Ministry of Education. Independent enterprises are rare. There are efforts of computer industries developing educational software for their equipment, software houses developing educational computer related material, and private schools using computers as a means to place their students in contact with the information era. (These students see the computer about 1 hour per week.) Public schools do not have computers and all the research concerning the use of computers in education is sponsored by government agencies. The Ministry of Education (MEC) has set up 5 research centers to study the use of computers in education (EDUCOM), promotes an educational software contest, and is setting up an educational computer center (CIED) in each State in Brazil.

The EDUCOM Project was set up in 1983. At this time all the universities and research centers in the country were invited to submit a proposal regarding the use of computers in education. The proposal could be about development of hardware, software, courseware, material, methodology, and evaluation of the effectiveness of the use of computers in education. It was required that the research center should work with a public school and the local university should be involved in this project. From the twenty-six proposals, only 5 were selected. The selection of these centers was done by recognized professionals in the field. Besides the quality of the proposal, other criteria included diversity of computers in education usage (CAI, Simulation, Logo, etc.) and geographic location of the center in order to cover all the different regions in Brazil. These centers are set up at:

- Federal University of Pernambuco (northeast region of Brazil.): the introduction of computer programming and computer concepts to students from the elementary and secondary schools. The first language used is Logo, although other computer languages and other tools are

being used.

- Federal University of Minas Gerais (central region): the development of computer assisted instruction for different subjects of secondary and high schools.
- Federal University of Rio de Janeiro (central east region): the development of simulation programs for secondary and high school students.
- State University of Campinas (central east region): the usage of Logo with elementary, secondary and high school students.
- Federal University of Rio Grande do Sul (south region): There are three EDUCOM subprojects. One is about the usage of Logo with elementary and secondary school students; the other about usage of authoring languages to develop instructional computer material, and the other about the development of an authoring system.

All of the EDUCOMs are concerned with the development of materials, training of people to support their particular activities, and the development of specific research topics related to pedagogical, cognitive and social issues.

The National Contest of Educational Software has taken place every year since 1986. The main purpose is to identify the most important educational software being developed in Brazil. It is opened to private companies, research groups and individuals producing software or tools related to the use of computers in education. During the 1986 and 1987 contests about sixty pieces of software were entered. All of them are described in a catalog published and distributed by MEC and the software products considered of better quality were given awards (about 10 each year.)

The Center of Informatics in Education (CIED) is an attempt to gradually introduce the computer into the public education system. Instead of placing one computer in every public school, the government's goal is to establish a computer center in every city along with model of the local library. The function of these centers is to serve students and teachers, to train teachers, and to disseminate throughout the community the different usages of computers. Thus, MEC, in collaboration with each State Education Secretary (twenty-six of them), is placing a CIED in each State (its actual location is a decision of each Secretary.) For this, MEC sponsored a 360 hour training course for 2 professionals of each State Education Secretary and is providing about 40 computers per CIED. To date there are 10 CIEDs installed in some State Education Secretaries have decided to install additional centers in their own states.

## Logo

It is fair to say that there is a Logo community in Brazil. This can be seen by the number of schools, research centers, as well as the number of teachers and parents requesting courses and information about the use of Logo.

## Hardware and Software

There are two versions of Logo in Portuguese available in Brazil. One version was developed by a Brazilian computer company, Itaotec, in collaboration with the State University of Caminas. Their terminology and error messages were carefully studied and tested with a large group of people. This version is available for the Itaotec CPM based computer and its PCXT-like computer. It is also available for the MSX computers produced by two different companies. This is the Logo language that has been used by the research centers and in the CIEs. There is another version available Apple-like computers. It is a translation of the MIT Logo version. However, the Portuguese version is not recompiled and the terms and error messages are not full Portuguese words and sentences. This Logo version is used by private schools that bought their computers to take advantage of the large number of educational software packages available for the Apple.

## Logo Users and Logo Material

Logo has been used by different populations: pre-school, elementary and secondary; students with learning difficulties, children from the slums, physically handicapped and deaf individuals, college students and adults.

About a dozen Logo books in Portuguese are available. Most of them are for the Apple version of Logo (Students in private schools using Apples can buy books.) Most of them are introductions to Logo for children and are restricted to Turtle Geometry. This has helped perpetuate the image that Logo is for children. In order to change this image the research centers are producing and disseminating materials about more advanced subjects in Logo. The community is ready now for a Brazilian Logo magazine or a Logo newsletter to distribute these materials and research results.

## Conferences

The lack of depth in Logo activities is quite noticeable in the articles presented at the two Logo conferences, one held in 1986 and the other in July of 1988. Most of the presentations lack the scientific approach and tend to be descriptions of simple Logo programs or descriptions of how the students like Logo. This situation probably reflects a phase which the Logo community in Brazil must go through. However, we are gaining momentum and practical experience which will enable us to take the next step.

## The Value of a Group Project

by Charles R. Mc Nerney

*[Editor's note: Although this article discusses a teacher training workshop, many of the ideas apply to the teaching of Logo to students of any age.]*

### Course Background:

The University of Northern Colorado offers four Apple Logo workshops during the school year. Two workshops are offered during the regular academic year, each ten weeks long, one three hour session per week. The summer workshops are of two week duration with one session per day. The sessions are three hours long with a fifteen minute break during each session. Each workshop is limited to thirty students, the size of the class being determined by both equipment and class management constraints. Presently, three students share the use of either an Apple Two Plus or an Apple IIe. This is not considered to be an ideal arrangement; two students to a computer is preferred.

The students enrolled in these workshops are, in general, teachers of grades K through 12, many of whom have never seen a microcomputer before. Most participants are not enrolled in a degree program; it appears they take the workshop because of personal interest and/or a sense of professionalism.

Because the course is a workshop, a grade of satisfactory is given for successful completion of the course requirements; a grade of unsatisfactory is given otherwise. Course requirements include the satisfactory completion of a group Logo project, a written critique of *Mindstorms* by S. Papert, and attendance at each of the workshop sessions. The workshop carries three quarter hours of graduate credit. The course content includes microcomputer orientation, use of disks to store and retrieve data, turtle graphics, procedures, recursion, workspace management, packages, use of the startup file, the Apple Tool Kit, and, the use of numbers, words, and lists. The course texts include Abelson's *Apple Logo*, MECC's *Apple Logo In The Classroom*, and Papert's *Mindstorms*.

### Course Conduct and the Group Project

The workshop consists of ten groups, three participants to a group. A group project is assigned the first session. This project is to be considered an ongoing project to be completed by the end of the ninth session. The project is the main focus of the workshop with the majority of the workshop time dedicated to the successful completion of the project. Each project is shown on the last session of the workshop with the group participants explaining to the other students how they developed their project and then answer-

## Group Project -- continued

ing any questions. Each group project is put on a master disk which is duplicated for each class participant. Thus, each student leaves the workshop with his project plus nine other projects.

It goes without saying that the initial reaction of the participants to the project assignment is one of disbelief, hesitation, fear, and a few other "well-chosen" adjectives. No explicit guidelines are given except that the project must be nontrivial and of interest to the group. The choice of the group project is probably the most frustrating part of the workshop for the participants. Class time is taken during the first session to present projects developed in previous workshops. Most of these projects fall into the realm of turtle graphics with scenes such as mountains, schoolhouses, and the like. Some projects include interactive components which combine graphics and nongraphics.

The workshop is structured in such a manner that at least one new Logo topic is presented at the beginning of each work session with follow up discovery and reinforcement activities. Time is provided for work on the project during each session. As the workshop progresses and more Logo tools become available to the group, the presentation and activity time diminishes while the project time increases.

### The Value of the Project

Seymour Papert describes Logo as a comprehensive environment in which the student can learn to think. Problem solving is of prime importance in the Logo environment and the process of learning is certainly as important as the product of learning.

The workshop project embodies much, if not most, of what a Logo environment experience should be. Bull and Tipps put forth this proposition when they state, "Effective Logo learning is project oriented." (see Dec. 83, *National Logo Exchange*) Students immerse themselves in a discovery environment whereby conjectures are put forth, tested, accepted, rejected, modified and/or expanded. The process of learning is meaningful since the process is a means to an end, namely, the finished project. Students interact with students; teacher and learner collaborate intellectually on a sharing basis; "new situations that neither teacher nor learner has seen before come up frequently and so the teacher does not have to pretend not to know" (see *Mindstorms*, p.115). In short, we see student-with-student, group-with-group, teacher-with-student, and teacher-with-group interactions to confront and solve problems as they arise.

The workshop project is ongoing. It gives the workshop both purpose and direction. The class activities provide tools to be used in completing the project which, in most cases, is

continually evolving. It is amazing to see a group start a somewhat simple project (although the group would hardly use the word "simple" at the outset), and then continually upgrade the project as their confidence and abilities improve. In one such project, a group initially decided upon a marsh scene, complete with pond, cattails, lily pads, and a frog. As the marsh scene evolved, it was decided that there should be a bug flying around the pond which would eventually be "captured" by the frog. The final project, produced by Carolyn Anderson, Ellen Foster, and Marcy Trahan, was very impressive with its rich array of colors and the "fly" buzzing around the pond before the frog's tongue shoots out to capture it. (See this month's cover.) It should be mentioned that the procedure for the buzzing fly was a clever adaptation of a similar procedure used by another workshop group to show a bee pollinating a flower.

Participants become completely engrossed in the completion of their projects. Success builds upon success. An intrinsic group pride emerges in that no group is competing against any other group for a grade. Course evaluations repeatedly indicate that the quality of the projects has little to do with the project being viewed as a course requirement or as a source of competition. Adjectives used to describe the course include "fun", "challenging", "rewarding", "worthwhile" and a host of others. Simply stated, the group projects achieve both the affective and cognitive objectives suggested by a Logo environment.

### Finally

The purpose of this article is to emphasize the importance of assigning a group project when Logo is taught in a workshop of adequate duration (projects such as I have depicted might be difficult to achieve in an abbreviated workshop of 1, 2 or 3 days). I have read articles about, and talked with teachers and participants in, Logo courses wherein the course content was the major focus of the courses. Discovery activities are, of course, incorporated in these courses as a means for teaching the content. However, the emphasis does not seem to be process-oriented. Perhaps group projects such as I have described may put a little more "process into the product."

Further information regarding the workshop can be obtained by writing to

Chuck Mc Nerney, Dept. of Mathematics and Applied  
Statistics, University of Northern Colorado,  
Greeley, CO 80639

# How to increase your Logo Power

Whether you're a Logo teacher, trainer, or enthusiast, you know that this powerful computer language has the potential to have a significant impact on how teachers teach and how students learn. ICCE's Special Interest Group for Logo Educators (SIGLogo) offers you a forum for the exchange of ideas, concepts and techniques.

**What is SIGLogo?** SIGLogo is a professional organization that helps Logo Educators get ahead. We sponsor meetings, conferences and workshops, providing a support community for Logo-using educators. Novice or expert, you will find helpful information in each issue of our journal, *Logo Exchange*.

**Satisfaction Guaranteed.** Whether you teach Logo or use Logo to teach, SIGLogo and *Logo Exchange* bring you a wealth of ideas from top Logo educators throughout the world, providing you with current information on Logo research, resources and methods. We're your personal window for professional Logo activities.

**Join SIGLogo Today!** As a member of SIGLogo, you will receive the *Logo Exchange* journal nine times per year. SIGLogo members are invited to participate in local, regional and national meetings and to contribute to the flow of ideas through *Logo Exchange*. *Logo Exchange* is published monthly except for June, July and August. SIGLogo membership is \$24.95 for ICCE members, \$29.95 for non-members.

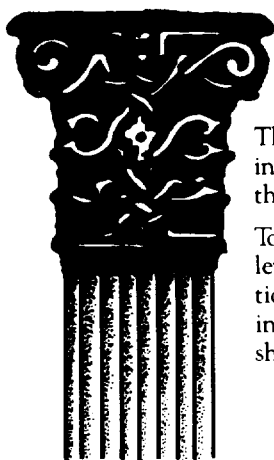
The International Council for Computers in Education (ICCE) is the leading U.S. and international professional organization for computer educators. It is non-profit, supported by more than 50 organizations of computer-using educators worldwide.



**Send For A Free Catalog Today!**

**ICCE/SIGLogo, University of Oregon, 1787 Agate St.  
Eugene, OR 97403-9905. Ph: 503/686-4414.**

# ICCE



## About ICCE

The International Council for Computers in Education was founded by Dr. David Moursund in 1979 as an organization that would foster appropriate instructional use of computers throughout the world.

Today ICCE is the largest professional organization for computer educators at the precollege level. It is nonprofit, supported by 14,000 individual members and more than 50 organizations of computer-using educators worldwide. These organizations are statewide or regionwide in scope, averaging 500 members each. Approximately 84% of ICCE's individual membership is in the United States, 12% is in Canada, and the remainder is scattered around the globe.

### About The Computing Teacher

ICCE publishes *The Computing Teacher* journal. *The Computing Teacher* provides accurate, responsible, and innovative information for educators, administrators, computer coordinators, and teacher educators. The journal addresses both beginning and advanced computer educators through feature articles, columns, software reviews, and new product and conference listings. Contributors to *The Computing Teacher* are leaders in their fields, consistently providing the latest information in computer education and applications.

### Publications, Special Interest Groups

In addition to *The Computing Teacher*, ICCE provides a number of publications to computer-using educators. ICCE's Special Interest Groups provide in-depth information for computer coordinators, teacher educators, computer science educators, and Logo-using educators. *C.A.L.L. Digest* is published nine times per year for ESL teachers. ICCE committees address a variety of ethical and practical issues important to the computer-educating community.

### Independent Study Courses

ICCE offers graduate-level independent study courses, designed to provide staff development and leadership. These courses have been approved by the College of Education at the University of Oregon and carry graduate credit from the Oregon State System of Higher Education. Participants correspond with instructors by mail.

Write for information and a free catalog today!

