

What is Logo?

Logo is a computer language which was developed to provide an environment which allows learning to take place as naturally as possible. Seymour Papert and his colleagues at Bolt Beranek and Newman and later at MIT set out to create a computer language which would combine the capabilities of artificial intelligence with the theories of Jean Piaget in order to allow a learner to build his own intellectual structures through estimation, interaction, experience and revision.

The Logo language is designed to provide an environment in which the child/learner is in charge of

- Setting a problem to solve.
- Making choices.
- Playing with the problem, experimenting and trying out solutions.
- Building on what he has already done to do something more.

The language is interactive. You learn it at the computer. After working with a problem for a while, you edit it, revise it, and then play your next steps.

With Logo, a young learner can enter directly into the world of turtle geometry. Without memorizing formulas, he can create procedures for drawing squares, triangles, and circles.

This is unlike my own experience in geometry. I learned geometry in high school only because I was required to take the course as preparation for college.

Geometric thinking is possible now for any learner without a series of pre-requisites. At the Lamplighter School in Texas, three year olds have used Logo to

Molly Watt

explore turtle geometry. And MIT students use the concepts of turtle geometry used in Logo as a way to explore mathematics.

Seymour Papert is often quoted as saying that "Logo has no threshold, no ceiling." I have heard it said that Logo is a six year old's dream and a computer scientist's nightmare. Logo makes complex explorations possible for learners of all ages, without imposing artificial hurdles.

With Logo, a young learner can enter directly into the world of turtle geometry.

How does a child explore turtle geometry? It is really quite simple. Every child knows how to move from one place to another. Using his own experience in walking, a child "teaches" the turtle to move across a computer monitor leaving a trail or line to create a drawing.

Let's call our learner Wendy. Her first experience might be something like this:

1. She decides to experiment by drawing a square.

2. She paces a square on the floor to notice how she draws it.

3. She remembers how she did it and types a set of commands to the turtle which might look like this:

FORWARD 25 (no, not far enough)
FORWARD 25 (okay, that's enough)

RIGHT 40 (oh, that wasn't what I meant at all, let's try more) RIGHT 40, (well that's not it, try again) RIGHT 10 (that's it) (let's see, what was it,) FORWARD 50
RIGHT 90

FORWARD 50

RIGHT 90

And so on until she has a square. With that accomplished, she can simplify the steps and teach the computer to do it by shifting to the edit mode. What shall the name of this procedure be?

Very often a student will name it with her own name, typing:

```
TO WENDY
FORWARD 50
RIGHT 90
FORWARD 50
RIGHT 90
FORWARD 50
RIGHT 90
FORWARD 50
END
```

The turtle will then "know" how to WENDY, and will execute a square every time the command WENDY is typed.

Now Wendy can enjoy playing with the procedure WENDY, and will probably type it over and over for the pleasure of watching the turtle "know" how to WENDY. The second WENDY will be drawn in a screen position which will surprise her, and our new Logo user will immediately type WENDY again just to see what happens. At the end of four

What is Logo, continued...

WENDYS, there is a new design which can be taught to the computer as a procedure containing the subprocedure WENDY. It looks a bit like a window, so Wendy can type:

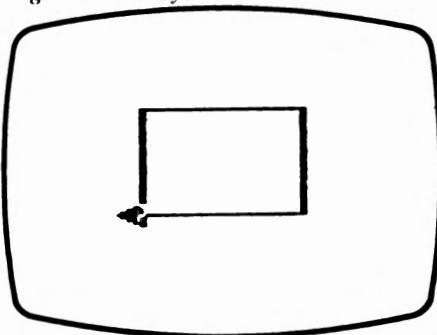
```
TO WINDOW
WENDY
WENDY
WENDY
WENDY
END
```

or

```
TO WINDOW
REPEAT 4 (WENDY)
END
```

Wendy has started the process of

Figure 1. Wendy.



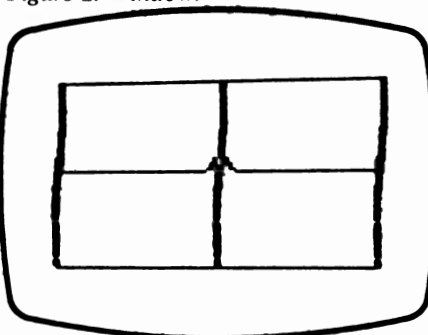
So it is with this first turtle drawing: Suddenly the Logo user is part of the world which assigns airplane reservations, cashes checks, and launches space shots. This is a world of power, which has in this first lesson become part of Wendy's own learning continuum. The process is intrinsically motivating and fascinating.

As every person can learn to use a pencil, everyone can — and will — learn to program a computer.

Projects by New Users

I think that it is important to look at some examples of tasks that some real students have set for themselves as problems to solve when beginning to learn Logo and to use a computer.

Figure 2. Window.



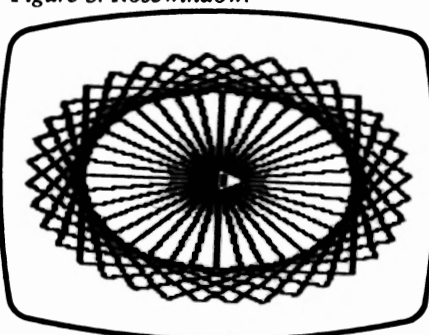
My purpose is to demonstrate the many working styles and interests which Logo can support. These students were in my classes in Amherst and Cambridge, MA and were between the ages of 8 and 17.

After several class sessions in a hands-on laboratory, I often suggest to my students, "See if you can make the turtle draw your initials." Many try the project, seem to enjoy the challenge, and then go on to some other experiment.

Two students, Ted and Stella, were working side by side, and became completely absorbed in drawing letters. Each determined to create a complete alphabet.

Ted immediately wrote a procedure to produce the whole alphabet, before creat-

Figure 3. Rosewindow.



- Learning to control the turtle.
- Using the computer as a tool.
- Teaching the computer to execute an original idea and naming it.
- Noting the steps she used in order to save them as a procedure.
- Using the procedure as a subprocedure in her second procedure.

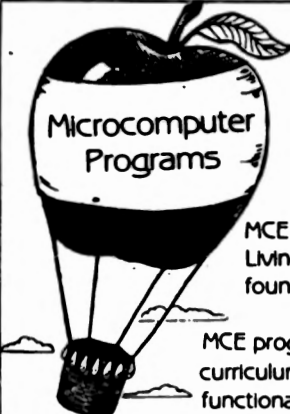
She has started to create her own computer language.

Taking the same example one step further, our new Logo user, in playing with WINDOW, may quickly discover that by using the procedure WINDOW and then rotating the turtle's position slightly a new design is made that holds a shape which begins to approximate a circle. This experiment can be formalized by typing, for example,

```
TO ROSEWINDOW
REPEAT 9 (WINDOW RIGHT 10)
END
```

As in all learning, the first job is to learn to control your tool. If you are using a pencil, you must practice holding it.

Your first written word may be your own name, which is a profound word to the writer. It permits you to show ownership, authorship, borrow library books, sign a check and send a greeting card through the mail. It is still a long jump to writing a dissertation or even applying for a job. However that one word puts you into the powerful world of words which make up magazines, recipes and phone books.



Above the Crowd

MCE offers you superior microcomputer programs in Basic Living Skills. These programs have been field tested and found effective.

MCE programs are sophisticated yet simple to use. They offer curriculum flexibility in special ed., voc. ed., home economics, functional math and consumer ed. among others.

Eighteen programs are available from \$44.95 for Apple II®. They feature: branching to individual reading and ability levels, a variety of instructional formats, vivid color graphics and advanced teacher control options.

To learn more about MCE programs call collect (616) 345-8681 or return the coupon below.



MCE INC.
157 S. Kalamazoo Mall
Kalamazoo, MI 49007

☐ Send MCE catalog
 ☐ Request demonstration

Name

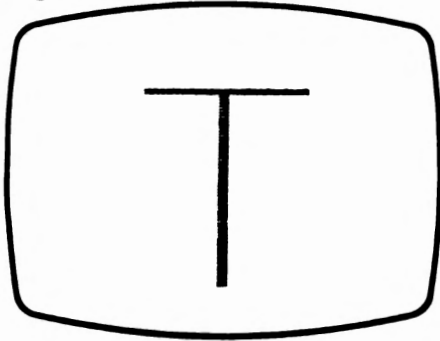
Address H10

City
 State
 Zip

Phone Best time

What is Logo, continued...

Figure 4. T.



ing the individual letters. It looked like this:

```
TO ALPHABET
A
DRAW
B
DRAW
C
DRAW
.
.
.
END
```

A B C

Ted then began the long process of creating procedures to draw each letter in turn. After his alphabet was completed he was dissatisfied with the amount of time each letter was displayed on the screen. He began to work on the more complex programming problem of creating something for the computer to do invisibly so that the letter on the screen could continue to be displayed for a longer period. This was a natural extension of his alphabet procedure.

Stella started out to solve what appeared to be the same challenge as Ted's. During the process, she became fascinated

Listing 1. Time.

```
TO WAIT :T
  IF :T = 0 STOP
  WAIT :T - 1
END

TO TIME :HH :MM :SS
  TEST :SS = 60
  IFT MAKE "SS :SS - 60 TIME :HH ( :MM + 1 ) :SS
  IFF MAKE "SS :SS + 5
  TEST :MM = 60
  IFT MAKE "MM :MM - 60 TIME ( :HH + 1 ) :MM :SS
  TEST :HH = 12
  IFT MAKE "HH :HH - 11
  WAIT 200
  PRINT []
  ( PRINT :HH :MM :SS )
  TIME :HH :MM :SS
END

"FALSE is 1
"TRUE is 1
```

by the proportions of height and width and how a variable input could change the relationships represented graphically.

The qualities of the computer which allowed her to experiment with inputs and then provided her with an almost instant reformulation of her letters heightened her excitement as well as her understanding about one way a computer could extend her thinking.

She began to collect patterns for monograms and initials which might make interesting designs on stationery. Her classmates used her program to think about embroidering on denim. For her final project, Stella created an art show using Logo procedures.

In one of the first Logo classes, Mark and Suzy became a working team. Mark seemed to need to be admired for his programming prowess and Suzy was reluctant to touch the computer.

This combination worried me because it seemed to reproduce the stereotyped, and I believe harmful, pattern of the way men and women relate to technology.

This twosome often seemed to have private jokes and to direct gestures toward classmates. Although I felt uncomfortable and wondered whether I should intervene, their classmates paid them no attention.

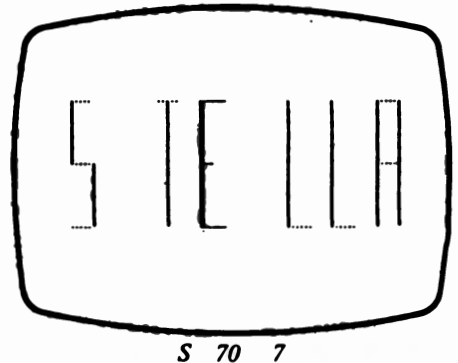
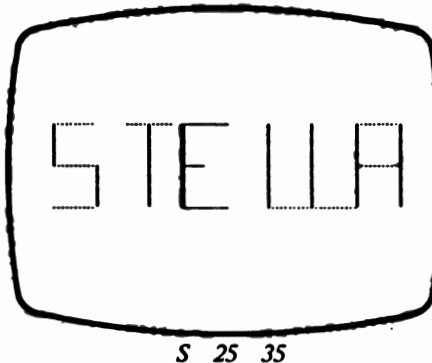
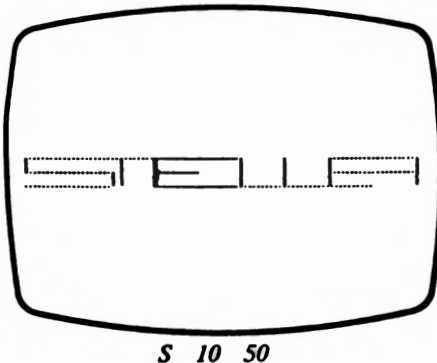
In the meantime, I watched and thought about them whenever I had a second to spare from this demanding programming class. I was aware that none of my suggestions had been accepted by either of them. While Mark was actively executing his ideas, Suzy admired, and I waited.

Then one day the duo split, each needing a computer to work on.

Mark was inventing a clock. It was a digital clock. The person using his program could type in the hour, minute and second and could then watch time move in five-second increments. His ability to capture some essence of time was satisfying to him, and truly mystifying to his classmates. Mark told them that his program was simply an approximation of time as is the time that all clocks keep.

Listing 1 shows how his program looked after a solid week of programming.

Figure 5.



What is Logo, continued...

His program does have some "bugs" which he has probably solved by now. One is the matter of conversion; the seconds do not convert to minutes after the number 60 is reached. This was not discouraging to Mark. This was the next part of his program which needed attention in order to make it work better.

On the other side of the classroom, Suzy sat at a computer alone. At first she seemed to me to be immersed in a science fiction book. As I moved nearer, I saw that she was using the science fiction book as a reference for her work. She was working with an intensity I had not seen previously.

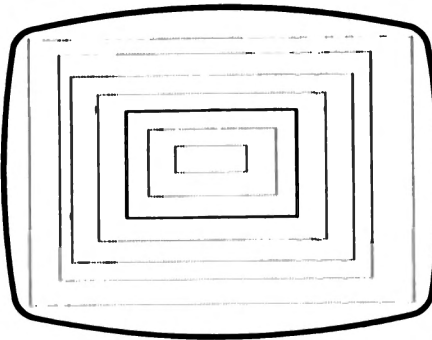
Between footstamps and fistshakes at the computer she was creating a science fiction book. She had discovered that the computer could print words. Linking her interest in reading science fiction with her limited knowledge of microcomputers, she had started on her first self-initiated project. She organized the procedures shown in Listing 2.

As the course ended, Suzy was beginning to experiment with turtle geometry to illustrate each chapter. My next steps for Suzy were to teach her how to save pictures drawn on the video screen, and to introduce her to the Logo text editor.

Suzy shared her project with her classmates during the last day of class. The science fiction aspect captured their imaginations, and they demonstrated genuine interest in her work.

With our visions of what is possible for a Logo user to accomplish, Suzy's example may not seem dramatic. I include it because I believe we often push students to

Figure 6. Door Through Time.



go too fast and forget the value of allowing a person to set his own goals. I do not believe that Suzy would have started sooner if I had structured the lessons differently; she needed a period of watching to get herself ready.

One visually oriented student spent many class periods creating drawings, saving his pictures directly on his disk without writing procedures. Jamie planned color and line relationships carefully and gave names which suggested to me the dimension of fantasy the computer offered him.

While classmates often gave pictures and procedures single letter names to avoid laborious typing, Jamie was content to take the time to type "Door Through Time," "Sparkle in the Night," "Experience in the Fifth Dimension," delighting classmates with his naming as much as with the pictures.

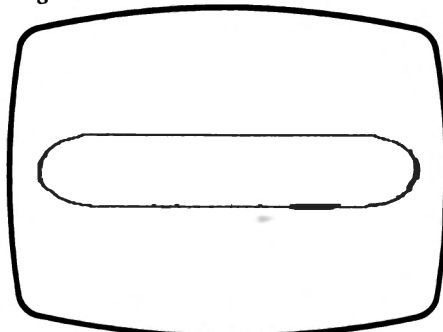
Another example of the way an individual entered the Logo environment or "mathland" is Jim. He simply started experimenting with the turtle.

He drew a shape which he later named "slot" because it looked like one. This slot with its circular ends and straight line connections intrigued him. During a series of lab periods, interrupted by several other projects, he progressed to "tslot" which added color and assumed a new position on the screen.

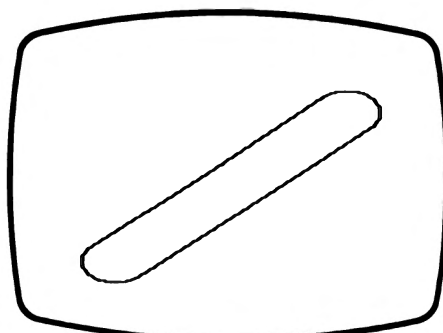
Next he played with variable inputs in order to experiment with size relationships. His moment of astonishment came when he discovered that the formula for a circle which he had memorized was approximate.

Jim was a serious student, and he often borrowed the Logo manual to read for homework. This provided him with many ideas about Logo primitives which might enable him to write more complex programs. He took himself beyond turtle

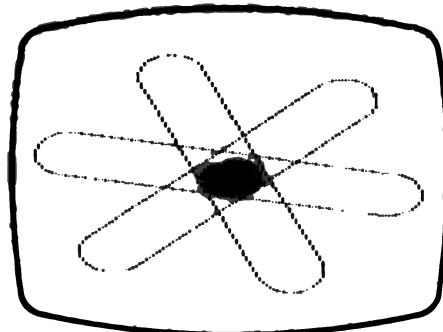
Figure 7.



Slot



TSlot



TSLOTT 2 20

Listing 2. Science Fiction.

```
TO MARS
  PRINT [A PRINCESS OF MARS]
  PRINT [THE GODS OF MARS]
  PRINT [THE WARLORD OF MARS]
  PRINT [THUVIA, MAID OF MARS]
  PRINT [THE CHESSMEN OF MARS]
  PRINT [THE MASTERMIND OF MARS]
  PRINT [A FIGHTING MAN OF MARS]
  PRINT [SWORDS OF MARS]
  PRINT [SYNTHETIC MEN OF MARS]
  PRINT [LLANA OF GATHOL]
  PRINT [JOHN CARTER OF MARS]
  PRINT [THIS IS THE MARS SERIES NO.1-11 IN ORDER
  BY EDGAR RICE BURROUGHS AUTHOR OF THE FAMEOUS
  TARZAN BOOKS]
END

TO DRAGONS
  PRINT [DRAGONFLIGHT]
  PRINT [DRAGONQUEST]
  PRINT [THE WHITE DRAGON]
  PRINT [THIS IS THE DRAGONRIDERS OF PERN TRILOGY
  BY ANN MCAFFERY]
END
```

What is Logo, continued...

geometry to write an interactive program which enabled the user to choose a polygon with any number of sides of user-specified length for the turtle to draw.

This project of Jim's pushed my programming ability, for I, too, was a new Logo user. I would often take my back-up copy of his disk home to work out various ways to solve his programming problems and bugs. After inventing or getting myself help with possible solutions or simpler examples, I created a new file called HELP to provide him with examples of possible solutions to his problem.

Jim, being an independent thinker, would study the examples in my HELP file, then say thoughtfully, "No, I don't believe that I want to do it that way," and continue with his own exploration and problem solving. I did notice that some of the HELP ideas found their way into his programming, and most of the inventions remained Jim's own.

At the end of the session, Jim was still working on centering his polygons above the text, and I had begun to use a HELP file with other students.

My first task was to invent a computer culture.

My last example is Manual. I have no file for his work. He was the class conceptualizer. He had started using computers because he enjoyed the fast action and energy of video games. He knew what would make an exciting program. Manual would strike up a conversation with a likely programmer describing an idea for a program. The talk continued as the other student moved toward a computer and started working out Manual's ideas. When Manual was satisfied that the person was "hooked" on the idea, he retreated and started another person on another idea.

I have Manual to thank for the work I do in the middle of the night, figuring out how to make a ball-shaped turtle appear to "bounce" off the edges of the screen.

These stories offer examples of the many ways in which students can explore a computer environment using the Logo Language.

Perhaps I've made it sound as if it "just happens." Because Logo permits a student to discover in a manner which is natural to the learning process it may appear that I have undervalued the teacher's role. Now we will look at some of the ways I have come to think about my role as teacher in a Logo classroom.

The Teacher's Role

Every computing class I have led has been made up of students with a wide

variety of experience and knowledge about computers. Some students have never seen a real computer. Some may even be certain that the computer is *the* enemy of humanity and spend energy directing prejudicial comments toward it. Other students may own their own computers, belong to "user's clubs" and have been programming for years.

This situation in itself was somewhat unnerving to me. I had made a commitment to myself to become knowledgeable about Logo and instructional uses of computers in order to integrate this tool into a school curriculum. I was determined to create a relationship between the current computer revolution and life in school.

I had no models. I had never used a computer nor seen any person other than my husband use one. My first task was to invent a computer culture. This culture needed to be able to support all the students *and* me as a learner alongside them.

Any teacher's first job is to assess the situation and equip himself with:

- Knowledge
- Experience
- Hardware
- Time

We must examine our own attitudes about computers in our society, acknowledge the areas creating distress and identify the areas bringing optimism.

As Logo teachers our roles will cover a diverse, and perhaps uncomfortable range including demonstrator, teacher/lecturer, teller, time structurer, problem setter, management solver, arbitrator, decision maker, challenger, helper, collaborator, process sharer, question asker, idea extender, observer, documenter, admirer, enjoyer, time provider, technician, and model learner.

I require each student to keep a journal of process notes, questions and descriptions of problems encountered. I read these regularly and respond.

In response I might write a comment which is similar to an oral response, "That sounds frustrating," "Wow! You figured it out!"; "I'd like to see how your procedure works"; "Jim figured out how to do a similar problem, why not ask him for help next time?"; "I don't know the answer to this question, let's get together after school and visit the computer store."

The journals provide a valuable vehicle in which to keep track of progress and to allow patterns to become visible. Often simply describing a problem will allow a student to understand it more fully, and thus be able to solve it. Journals provide direct access to help. They enable students to formalize their own thinking.

The journals provide me with a sense of being in charge, of knowing what is going on, and, a means of keeping records of student work. They provide an opportunity for a personal relationship with each

student on a daily basis. Since I am also learning Logo, the journals provide me with a sense of comfort; I can see what I need to learn, and decide what my own homework will be.

Beyond this, the journals provide both the students and me with an assurance that this is a collaborative learning experience, and that I am working *with* them. Confidence about this alleviates, I believe, for both the students and me the sense of anxiety which might otherwise be present in teaching and learning this subject. Most educators agree that anxiety interferes with learning.

As a Logo programming teacher, it is my job to make back-up copies of all my students' disks in order to protect the students from work loss due to damage or filing mistakes and to enable me to see the patterns in the work of individuals so I can plan my next formal lesson for the class.

Journals provide direct access to help. They enable students to formalize their own thinking.

I can collect small groups of students around a common interest or programming problem. Sometimes I will ask a student to share some work with the whole group as a teaching example or as a model. By examining the work on the back-up disks I can also determine whether a programming problem should be solved with a "gift" of a tool which the programmer is probably not yet ready to invent on his own. (Some tools I have given include procedures for creating circles, explosions, and countdowns.)

The disks give me time to work on programming problems by *trying* the program, and trying several solutions, away from the stress of a class period. Inevitably the disks force me to think about my own next learning steps.

During the actual lab time I wander, watch, listen, and answer.

I feel that the words I use are important.

Instead of solving a problem for a youngster by telling or showing the solution immediately, I usually say: "describe the problem," "Tell me what happens", "What did you want to have happen?"; or "Try it now and show me." Some teachers ask the student to "Teach me what you did."

This type of response is important for several reasons.

- It gives me, as teacher, time.
- It gives the student time.

What is Logo, continued...

- Description is a matter-of-fact task which can diminish emotion and allow the describer to see what actually happened clearly. (Frequently in the middle of describing, a student will say "Oh, never mind, I see what I did.")

If after a student has described the problem and neither of us knows how to solve it, we write a plan together in plain English words. It includes a statement describing what the student *wanted* to have happen. Usually when a solution is not clear to either of us, we are working with an example which is too difficult for us. We substitute a simpler problem "for practice."

Then, together we write a superprocedure in the same way that Ted wrote one for his alphabet, before he had invented the subprocedures. I make very sure that the first step in the procedure is one that the student and I can solve successfully right then.

Other jobs for the teacher include collecting, displaying and identifying resources.

I have rarely seen a computer which is not surrounded by a group interacting with it and kibbitzing with each other in a most congenial way.

I use bulletin boards to stimulate the learning process. I might post a weekly "Mystery Procedure," a new command with its definition and examples of its use, a challenging programming idea, a procedure to copy, a picture of a student's procedure, or a chart of students' names indicating their specific areas of expertise in order to make peer tutoring possible for every student. Sometimes I post an interactive program for students to copy, use, then modify and make their own.

Creating a Supportive Community

The computer community which we establish in our classrooms is for many students (and teachers) a first computer culture. As educators, we must be concerned about the values which are formed and used by this community of learners.

Many question my use of the word "community" in connection with computers. However, I have rarely seen a computer which is not surrounded by a group interacting with it and kibbitzing with each other in a most congenial way.

Most computers are a social and an in-

tellectual center in a classroom unless usage is specifically regulated otherwise. Watchers see new ways to solve problems, participate in brainstorming new ideas and derive a great deal of pleasure from the process.

Teena Crowley, a third grade teacher, said, "I wish Logo were everyone's introduction to community work. Everyone is involved and offers input from the start."

Usually during a first class with a group I find it important to declare myself a Logo learner as well as a Logo teacher. I do it because it is honest and because it establishes a basis for collaboration. The responsibility for helping and teaching and creating this class belongs to all. It is an ideal opportunity for me to model the motto "Life Long Learning," certainly an important part of any curriculum.

Many issues emerge in the establishment of a Logo community. One of the first is the issue of ownership of procedures. Is it copying or stealing to save work which did not necessarily originate from the saver's own inspiration?

The nature of Logo filing and saving makes it almost impossible for beginning users to keep separate files, and so from the first time SAVE is typed, a file of mixed-up origin exists.

I prefer to be excited by the possibilities of this "sharing" rather than regret it as a necessary "evil." I legitimize the sharing, giving, changing, and using of one another's procedures as part of the context of the community we are creating together.

I share my work with my students and I encourage them to exchange procedures among themselves.

Classes soon begin to create a vocabulary specific to their community. In one class a procedure named WING — basically an outwardly spirally triangle which appeared to fly across the screen — became a favorite expression for a way of moving quickly across a space with arms rotating rhythmically. This reference point created a sense of cohesion and inclusion among classmates and added a dimension of good-natured humor.

I credit some of my success in creating a supportive community to my insistence that part of the experience in computer class is participating in discussions about ethical and responsible computer use.

I am genuinely worried about how our society makes decisions about computer use. I make clear to my students that computers are surrounded by a human culture with developed values and a sense of conscious choice about their use.

I believe that one way people develop into responsible, rational human beings is by participating in discussions about dilemmas with peers.

I often start a discussion with a simple story. Ideally, it describes an issue which is real or potential issue for this class, with-

out embarrassing any group member. Once I told about a filing mistake I made when making a back-up copy of Ted's alphabet. Half the procedures for creating letters disappeared through my mistake. I had to tell him what I had done to his work, and how sorry I was to have made such a silly and harmful mistake.

Part of the experience in computer class is participating in discussions about ethical and responsible computer use.

A 20-minute period of sharing of mistake stories followed. No one was required to talk, but all class members were expected to participate by being part of our circle. The rules also prevented comments or judgments being directed towards any person or point of view. Discussion was encouraged.

Other issues which have worked for discussions in my classes are:

- Scheduling of computer use.
- What about someone who doesn't take a turn?
- Is it okay to borrow disks, copy disks, change disks?
- What about using "bad" or "dirty" words to name procedures?
- Should all schools have computers?
- Should all kids have to learn to program a computer?
- Why would a girl want to use a computer?
- What kinds of information should a school computer keep?
- Is it possible to have a really "private" computer file?
- What are some ways computers work in our lives?
- How can I get my family to value my programming instead of dismissing me as a "brain"?

The discussion which touched me the most deeply was requested by my students at the end of a summer school session. The subject was: now that we know so much about computer programming, how can we help our teachers feel comfortable knowing less than we do, so we can have computers in schools?

Beyond these ways of thinking about structuring a class and a curriculum, there are many opportunities for collaborative programming. One teacher has helped her students create a class adventure game using Logo.

The class planned the rooms, and then pairs or individuals created the procedures

October 1982 • Creative Computing

to make each room. Her role was to set up the structure and aid in the process of linking.

Harold Abelson's manual for Logo gives many examples of interactive programs which can be enjoyed by a class. Many teachers are using it to create their own programs for guessing numbers and creating crazy sentences.

Dan Watt's work creating curriculum for a dynamic turtle which moves according to laws of physics is interesting to use and then modify.

A great need exists for all Logo teachers to share their emerging curricula. This will enhance our collective understanding and provide more models about the ways students can work within a Logo environment.

What Logo Teachers Say They Teach

- Computer literacy.
- The history and learning theory in Logo.
- How to program a computer-like character, a turtle robot or Bit Trak.
- Controlling a turtle on a screen.
- How to pace out shapes and then teach the turtle to draw shapes.
- How to change pencolors and background colors.
- How to edit.
- How to initialize a disk.
- How to draw initials.
- How to use repeat.
- How to use subprocedures in procedures.
- How to use recursion.
- How to read a print-out of programs.

- How to draw procedure trees.
- How to use variables.
- How to use the Logo Manual.
- How to manage files and clear the workplace.
- How to use existing interactive programs and modify them.

This simple and straightforward list may be more or less what you expected. However when I ask my Logo students, whether school children or professional educators, what they learned in Logo class the list is quite different — fuller and more profound.

What Logo Students Say They Learn

- About problem solving and estimation.
- About thinking and learning styles.

**This person used to
develop applications
for the 8086/8088
market with the
best of them.**



SOFTech
MICROSYSTEMS

What Is Logo, continued...

- About how to use their own learning style.
- To think logically.
- To work without emotional manipulations; the computer doesn't care whether you feel angry.
- To use procedural thinking.
- To use strategies for problem solving.
- To become comfortable thinking mathematically.
- To be able to think geometrically.
- To be able to consider laws of motion.
- About language by creating my own system for naming procedures.
- About graphics and design.
- How important revision of procedures and text is, and how simple it is to do.
- That decimals are useful.
- How to type.

- How to be patient.
- How to take risks in working.

The most significant things they claim to have learned include:

- Looking at their own mistakes with an interest in understanding what happened instead of shame.
- Feeling competent in setting their own problems and supported in solving them.
- Understanding that learning and doing involve frustration and ease; they go quickly or slowly, parts are intriguing or boring, and this is what makes up all work and life.

There is no question in my mind that working with computers is one way of forming a direct link with a sense of the future. Computers can provide a sense of

optimism, new frontiers to be explored, and a kingdom to be conquered.

Logo was developed to create an interactive environment, a mathland, in which students could set their own pace, problems and goals. It is a comfortable way for me to enter the future. It is a challenging way and it is a way that I can understand, choose and control. □



**This person
used to develop
applications for
the Z-80 market
with the best
of them.**



SOFTech
MICROSYSTEMS