ISSN 0734-1717

The National LOGO Exchange

Volume 1 Number 1 FORWARD 100! September 1982

Mindstorms — Don't Miss It

If there is to be a classic book about LOGO, it is sure to be Mindstorms by Seymour Papert. It is practically impossible to know anything about LOGO without having heard something about Mindstorms, since Dr. Papert was one of LOGO's founding fathers some twelve years ago.

If you have not yet read this book, don't miss it. The essence of the LOGO learning philosophy is contained therein, written in an easy, straightforward style. But that is not all that awaits you.

Papert develops many powerful ideas which are of importance and concern in today's education systems. Most deal directly with the role of computers in the classrooms.

His main thrust seems to be in the direction of "natural learning," a philosophy he gained from direct study with Piaget. He views LOGO as an ideal vehicle to provide the children with a learning environment in which they can explore, experiment, and play, using the computer and its capabilities in self-directed ways.

He also looks to the future, and sees the need for a revolution in education as we know it today! The QWERTY phenomenon is his name given to any process which is so deeply established that, regardless of any suggested improvements or developments, no changes are permitted. "We have always done it this way," is the motto of the QWERTY-ite. Does this idea sound familiar? Perhaps you will find this part of Mindstorms extremely stimulating!

If you are looking for LOGO lesson plans, you will not find them in this book. But, you will find several concete ideas and specific philosophical concepts upon which you can easily base your LOGO plans, activities, and curricula. And you will gain an appreciation for the incredible potential of LOGO, the next giant step in education!

Mindstorms: Children, Computers, and Powerful Ideas, Seymour Papert, Basic Books, New York, 1980.

LOGO Reinforces Geometric TABS Skills

by
Linda Nix

The computer was a new addition to our school this past year, and, as the fifth grade teacher, I was anxious to use it as much as possible as an extension of my regular teaching strategies. In particular, I wanted to use it to reinforce required TABS skills. I found LOGO to be an exciting and effective way to develop recognition of geometric shapes.

My technique was for students to recreate geometric shapes by using LOGO and to record their programs in a notebook for future use by other students. Since LOGO was new to both my students and me, I started slowly with a teacher-directed activity demonstrating computer commands in basic LOGO for the turtle mode. The students sat on the floor in front of the computer and took turns giving directions to the computer, which I typed in, to draw a particular shape.

Several students, because of their scounting activities, quickly saw the relationship between the degrees on a compass and the size of the turns required to make the correct angles.

After drawing a square, a rectangle, and a triangle, we drew the outline of a house. It was exciting to observe the group interaction that took place. Even the shyest child became involved.

The students were then paired with partners and given the assignment to write directions for the computer for other geometric shapes. They worked at their desks using paper and pencil. Some used a compass to help them decide on the size turns to have the computer make.

During this time, it was interesting to watch the decision making process that was taking place and the student interaction. Many students worked together beautifully. For others, it became a test of patience and self-control.

continued on page 4

The National LOGO Exchange, copyright 1982 Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly September through May, $25 per year, $45 for two years, mailed FIRST CLASS from Charlottesville, VA. Add $5 per year for addresses outside the United States. Single copy price $3. Opinions expressed by the authors are not necessarily those of The National LOGO Exchange. Address all editorial correspondence to the editor at The National LOGO Exchange, Post Office Box 5341, Charlottesville, VA 22905. ISSN 0734-1717

Editor.....................Tom Lough

WELCOME!

Welcome to The National LOGO Exchange, a monthly newsletter for LOGO teachers and parents! We hope that you will be able to make use of the tips and techniques which are presented to you here.

In each issue, we will have columns to help you develop LOGO plans and activities. Book reviews and critiques will be included, as well as comments on LOGO products we have reviewed. Important features will be articles written by classroom teachers and parents, reporting on their LOGO activities and discoveries.

We also hope that you will be willing to share _your_ experiences and lessons in LOGO with your fellow readers. The future for LOGO is so exciting! Here is an innovation which is capable of changing education as we know it today. How its use develops will determine its effectiveness, to a large extent. You are in a position to influence that development by reporting your LOGO successes to others so their children can also benefit. Why not write your LOGO article today?

Here at The National LOGO Exchange, our motto is FORWARD 100! This reflects our enthusiasm toward LOGO and its role in education. We are very excited about the potential represented by LOGO. We want to press on in a FORWARD direction with our efforts to bring LOGO to a position from which it can influence our country's children. We want to give 100% support to you, the LOGO teachers and parents, as you work with LOGO in the classrooms and homes. Please feel free to adopt this motto as your own, for LOGO, and for life!

FORWARD 100!

Tom Lough
Editor

TIPPS for TEACHERS

by
Steve Tipps

Distance and Direction

The first four primitive commands which are introduced to the new LOGO programmer are FORWARD, BACK, RIGHT, and LEFT. With these commmands, the turtle can be directed to move in many directions and draw an infinite number of figures. However, the four commands also are codes for the first two concepts which are embedded in the turtle's microworld: distance and direction.

Distance (how far) and direction (which way) are two different ideas which children and adults need to recognize as separate ideas if they are to achieve an understanding of the geometric and arithmetic rules which govern the turtle. These concepts need to be introduced at the body and toy level before the computer and screen turtle are used. A number of simple direction games common in preschool and primary classrooms can be adapted for use.

Body games. "Simon Says" and "Mother, May I?" are the two most common variations of body direction games. Simon says, "Take two giant steps," is easily changed to "Take two turtle steps." I usually emphasize that turtles take rather small steps, so that even 10 turtle steps would probably not equal one giant step. Children can also walk off the dimensions of the room or the hallway using turtle steps. This activity is also used as an introduction to measurement and units.

The emphasis has been on distance and, in particular, the command FORWARD. However, BACK commands can also be used in "Mother, May I?" and in measuring distance. Many times, children will want to turn around and step FORWARD when given the instruction BACK. The teacher would emphasize that, for both FORWARD and BACK instructions, the child's face is pointing the same direction. This provides an opportunity to introduce turning or direction as a separate concept.

continued on page 3

In the kindergarten and primary grades, teachers usually teach left and right as labels for hands and feet. LEFT and RIGHT commands can be incorporated directly into the body games. "Turn 10 turtle turns to the LEFT," or "Make a giant turn to the RIGHT," gives reasons to know and ways to use directions. Children can argue about and define the size of a giant turn. Coming to a class agreement about the size is in keeping with the child centered philosophy of LOGO.

Toy games. When the concepts of distance and direction are established, they can be used in a variety of ways. "Control Tower" is one activity which my students always liked. One student would leave the room while the others would rearrange the desks into a maze and a landing strip. The task of the Air Traffic Controller or Committee was to guide the blindfolded student by giving commands of FORWARD, RIGHT, or LEFT. (BACK is used with a helicopter, but students will not allow it with an airplane!) The object of the blindfolded pilot, of course, is to follow the directions carefully and quickly to bring the jet down on the runway. Whether there was actually a toy airplane or not, children enjoyed the mock situation, and would invent a variety of dramatic reasons for the pilot and plane being disabled.

Variations with automobiles, submarines, and helicopters are possible. Routes and roads can be made out of blocks. Maps of the school and routes from home to school or to the grocery store can be developed. "Mystery Places" for third and fourth grades are a good class puzzle. One or two children can describe a route from the school to an unknown destination by counting blocks and turns. Similar activities should be part of mapping units in social studies, science, or math for the primary grades. In each case, the differences between distance and direction are the primary conceptual learnings.

If the teacher has access to a floor turtle (available from Terrapin Inc., 678 Massachusetts Avenue, Cambridge, MA 02139), the direct transfer to the computer turtle is apparent. However, if no floor turtle is available, the intermediate steps with objects should not be skepped. On a large sheet of manila or graph paper, students can draw figures and designs which are dictated to them. Different magic markers can be used. While this is a crude simulation of a floor turtle, the drawing is a good representation of what will be done with the screen turtle.

continued

Screen games. Before students get to the computer and type in FORWARD, BACK, RIGHT, or LEFT, I am convinced that they need adequate time to explore the ideas of distance and direction at the more concrete levels of the body and the toy. The first problem which students (children or adults) encounter is recognizing that a RIGHT or LEFT command does not cause the turtle to draw a line. The amount of time which is spent being mystified seems to be related to the ability to convert the commands into a more concrete level and see that direction commands never cause any movement. The amount of time which needs to be spent at the more concrete levels may vary from individual to individual and age to age, but it is a foundation for the thinking and debugging process of LOGO. Without the foundation and process, I have observed learners swear that the only place where the turtle can draw a square is in the center of the screen! Full control of the turtle is based on a full understanding of the concepts of direction and distance.

A number of screen games can be played or invented to increase awareness and facility of control. One game is "Hit My Finger," in which one child puts a finger on the screen and the other tries to guide the turtle to it. A lot of experimentation will go on as the pointer tries to find more difficult places to put a finger, and the seeker tries to cut down the number of separate moves needed to touch. The exact rules of success can be left to the ingenuity of the children (who, by the way, will usually set increasingly exact standards!). A variation is to put colored adhesive dots on the screen for children to reach. A declaration of "I can hit the red dot in four moves," may be the beginning of a bidding process which children soon find has a limit of two moves, except for certain places on the screen.

The teacher can change the game by establishing a procedure called SPIN (listed elsewhere in this issue) which randomly sets the direction of the turtle at HOME. (This brings up the question of whether or not a teacher should ever preprogram anything for children. This important issue has many thorns which will be dealt with in later columns. In this case, I feel that, since the purpose of the game is to increase understanding of distance and direction, this small preprogram serves that instructional purpose.) A more complex distance and direction procedure is suggested in the Solomon

continued on page 8

TABS continued

I then posted a schedule of times the computer would be available to each group to test their programs. The students could sign up for one class period, before or after school times, or during recess.

During this entire testing of programs, regular math assignments were continued. After the programs were fully tested, the students recorded them in a notebook for future reference. Several of the students wrote programs containing more than one geometric shape. Later in the year, students shared their programs with other grade levels experimenting with the LOGO language.

This coming school year, I plan to expand this activity a step further by doing the exercise earlier and saving the students' programs on a diskette. Later, the students can run the programs as an exercise to review recognition of geometric shapes prior to testing.

As the year progressed, it was exciting to watch the students experiment and discover interesting ways of making the computer do what they wanted, using turtle and sprite modes. They developed an awareness and appreciation of what was involved in making the computer games they love to play. It reinforced my opinion that LOGO can be an exciting and valid addition to an elementary math computer curriculum.

Linda Nix is a fifth grade teacher at Forest North Elementary School in the Round Rock Independent School District, Austin, Texas.

SPIN

Here's a procedure to set the turtle to a random heading no matter where it is located. This could be used in many introductory exercises, as suggested in the Tipps For Teachers column of this issue. I'm confident that you could devise many more creative uses for SPIN.

If you or your students work with SPIN and achieve some interesting results, write them up for publication! It is through our sharing that our children benefit.

For the Apple:

```
TO SPIN
  SETHEADING RANDOM 360
  END
```

continued on page 8

You're Invited!

The avalanche is beginning! LOGO is on the way ... and it has the potential to revolutionize education as we know it!

But, the teachers across the country need your help. They are beginning to experience challenges which you already have met and fears which you already have faced down. Would you like an opportunity to be of service to untold numbers of fellow LOGO teachers and their students? If so, then consider writing a short article about an incident from your LOGO experience for The National LOGO Exchange.

This monthly newsletter presents articles written by LOGO teachers for LOGO teachers. We are looking for short (500 words or less), snappy, practical pieces about specific LOGO incidents from which you learned about yourself and how you were teaching. Do you have a neat technique or method for encouraging children to explore a certain concept? What was your very first day with LOGO and children like? "If I had it to do all over again, I would be sure to ... " What are your favorite programs developed by your students? How did you help them discover the difference between :variable and "variable?

You could double the number of such questions in less then 15 seconds, I'm sure. Why not take a few minutes to write about one or two, or more?

And please do not limit yourself necessarily to those things which worked. The beginning LOGO teachers need to know what things might <u>not</u> work as well!

Parents, we need to hear from you, too! What are you experiencing with LOGO? What are your concerns? When you observe how your child is thinking when working with LOGO, what are your thoughts? Do you have any ideas or suggestions which teachers might find helpful for their LOGO work?

Mail your articles to The National LOGO Exchange, Box 5341, Charlottesville, VA 22905.

We thank you and look forward to hearing from you. Your fellow LOGO teachers thank you and look forward to reading about you. The children thank you and look forward to benefitting from you.

FORWARD 100!

MICROWORLDS

by

Glen Bull

Design of a Language

Consider the following:

- An educator suggests that a subset of BASIC (eight - command BASIC) may be a more effective introduction to programming for children than LOGO.

- A salesman in a computer store tells a customer that a version of Pilot with turtle graphics is "LOGO."

- A graphics language with limited test-handling capabilities is described as a new implementation of "LOGO" in an article in a national publication.

The three occurrences described above have a common element in that they all betray fundamental misconceptions of the true nature and intent of LOGO:

- LOGO is better suited to educational use than BASIC because of the structure of the language, rather than because there are fewer commands to learn.

- LOGO is more than turtle graphics. Turtle graphics is a highly visible feature of LOGO which has influenced development of graphics capabilities in a number of other languages, including dialects of Pascal, Pilot, and Smalltalk. Even so, powerful operators for manipulation of language are also an integral part of LOGO.

Papert argues that failure of educators to consider the design of educational languages will inevitably result in inappropriate structures, by default. The danger seen is not that the wrong choices will be made. Rather, the problem lies in the shaping of educational philosophies through lack of awareness that there are choices to be made

continued

The structure and characteristics of LOGO are not mystical, ineffable qualities beyond discussion. They are straightforward design specifications subject to examination. To do otherwise is to act from blind faith or religious fervor.

The most important characteristics which might be desired in a language intended for use with children can be summed up in three words: interactive, procedural, and extensible. BASIC was the first language designed to teach programming which was used interactively on a widespread basis. The importance of this advance would be difficult to overestimate. When centralized computing resources are expensive and scarce, they can be optimized by making people wait in a queue for processing time. Under those conditions, it may be minutes or hours before the results of a program are seen.

On the other hand, direct access to the computer and immediate results undeniably have a strong effect on learning. This philosophy suggests that a person's time is more valuable than a computer's. This may seem obvious now. At the time BASIC was developed, it was a suspect, controversial idea. It would be over a decade before the term "personal computer" was needed, in the sense of a computer inexpensive enough for exclusive use by a single person.

The wide-spread use of BASIC made it evident that it was highly desirable for non-professional users of computers to employ the computer in an interactive fashion. After a time, it also became apparent that BASIC was not the ultimate, perfect language. In fact, computer scientists concluded that the structure of languages such as BASIC fostered development of poor programming habits and error-ridden programs.

There are a number of reasons this is so. An important one is the fact that BASIC is not procedural, in the true sense of the word. Its closest approximation to a procedure, the subroutine, falls short in several respects. In a programming language, a procedure is an independent description of a process which performs an action. Once developed, this description can be treated as a black box by the remainder of the program. The mechanics of how the goal is accomplished inside the black box can be disregarded.

continued on page 6

Microworlds continued

The best-known procedural language developed for teaching programming is Pascal. It has almost universally been adopted for teaching programming by computer science departments. Pascal was developed for college students. Hence, a considerable knowledge of computer science is required to write even a short program in Pascal. Also, Pascal programs cannot be run as soon as they are typed, but must be compiled first. The benefit of Pascal is that it does encourage good programming practice, even when the user moves on to other languages.

LOGO is both interactive and procedural. A procedure can be run as soon as it is typed. A design goal of LOGO was an educational language with no threshold and no ceiling. In a certain sense this goal has been met, since LOGO has been successfully used by children in elementary school and by physics students at MIT. In passing, it might be mentioned that the common impression that LOGO is only of interest to children is incorrect. LOGO has facilities for developing routines in assembly language, as well as list processing capabilities found in few languages.

In LOGO, a procedure is invoked by naming. As a consequence, the langauge is extensible. This means that the user can develop new commands in LOGO, and, in effect, write a new language tailored to a particular need or application. This capability is only available in a few other languages, such as Forth. For example, LOGO does not provide a FOR ... TO control structure similar to those found in BASIC and Pascal. The philosophical emphasis of LOGO seems to be on recursive rather than iterative control structures. However, a FOR ... TO command can easily be developed in LOGO, if there is a need for this structure.

In its current implementations LOGO is, unsurprisingly, not a perfect language. The intended application of a language heavily influences its efficiency. LOGO is not extremely fast. Because it is a sophisticated language, it requires more memory than languages such as BASIC. In a classroom environment speed of execution and efficient use of memory are usually not the most crucial considerations.

The price of silicon computer chips will probably fall rapidly, and their capabilities will increase. As this occurs, limitations of speed and memory will become less of a factor. However, language characteristics which are artifacts of technologic limitations will continue to be perpetuated unless thought is given to desirable traits. For example, usually the absence of procedures

continued on page 8

Tell Congress: We Want Computers!

Those of you who are participating in the computer revolution will have a particularly important reason for making your presence known. There are two actions in Congress which need your support: The Technology Education Act of 1982 (HR 5573 and S2281) and The Family Opportunity Act (HR 6397).

The Technology Education Act of 1982 is of special importance to schools. It provides incentives for the major computer manufacturers to donate a state-of-the-art computer to individual schools. With only a few hundred thousand computers in schools to date, this bill, if passed into law, will provide a veritable avalanche of powerful computers for many schools, particularly for those less gifted financially (are there any otherwise?). But it is up to the legislators, our senators and congressmen, to vote on the measure. If we will take the few minutes necessary to write a note outlining our support of the bill, we will be doing our part to help make it work.

If you are a teacher, why not check with your principal to see if you could send in your letter on your school's stationery? How about getting the parents' organization involved? They could work on gathering signatures on petitions. Do your students want more computers in their schools? Let them participate in the democratic process. Writing a letter to Congress makes a dynamite social studies and English assignment!

The other bill, The Family Opportunity Act, provides for a tax credit to a family for the purchase of a computer system and software. This will encourage families who can afford computers, and who have just been "waiting to see what will happen," to take the plunge now. This will put computers within the reach of many children during non-school hours. Think of the growth opportunities this might provide!

In the background of all these remarks, of course, are tremendous implications for LOGO. Just think of all the gateways these two bills could open for LOGO activities if they become law! Our education system could experience a growth like nothing which has been experienced to date. Imagine what it would do for our children if nearly all of them would have the opportunity to grow up learning with LOGO! Don't you feel the children are ready for it? Don't they deserve it?

LOGO Your Boat

For those of you involved with physics, here are a few things LOGO to think about when dealing with a particular class of vector problems.

Remember the situation of a boat and a stream? In general, the problem is to figure out how to paddle the boat across the stream and arrive at a particular point on the opposite bank, knowing the speed of the stream and the boat.

Most physics texts encourage the student to figure out the constant angle at which to head the boat upstream, so that the vector sum of the speed of the boat and that of the stream will be a vector which points directly across the stream to the desired destination.

This is a correct approach, as far as it goes. However, as those of you who actually have paddled a boat or canoe know, keeping a constant angle with respect to the current while vigorously paddling is difficult to do in practice.

Rather, the easiest thing to do is to keep paddling the boat toward a certain distant point. If that point is your ultimate destination, you might eventually get there.

LOGO can be used readily to study this situation. Depending on the speed of the stream and the boat, it is possible to come extremely close to your destination. However, for choices much different from a particular range of combinations, you might end up quite a distance downstream, depending on your definition of "arriving at the other side."

The LOGO listing is this article is presented to you for exploration. Feel free to vary the parameters, the dimensions, and the test for arrival. The shapes of the various paths of the boat might hold some surprises for you. If you are a teacher, you might want to make the WATERCROSS procedure available to your students for informal research.

Initally, a stream is set up on the screen. The left and right banks are shown as vertical lines, and the "perfect path" to the destination is shown as the horizontal straight line. The procedure RIVER draws all of this and then places the turtle on the left bank facing the POST on the opposite side of the stream.

When the CROSS procedure begins, the boat (turtle) is paddled FORWARD for one increment of time at the BOATSPEED. Then the stream takes a turn, pushing the boat downstream (towards the top of the screen) at the STREAMSPEED for one increment of time. The heading of the boat is then

continued

reset toward the destination (SETHEADING TOWARDS :POST). Testing of arrival is performed next, with the initial criterion being to be within 2 units of the opposite shore. Then, the CROSS procedure calls itself again.

When the test determines that "you have arrived," the distance you are located downstream of the POST is printed at the bottom of the screen, along with a recap of the boat and stream speed.

As mentioned before, you can get some interesting results by varying the test criterion. Instead of 2, for example, try smaller and larger numbers.

The listing below is written for the LCSI Apple LOGO. Changing over to other versions should not be too much of a problem, I hope.

```
    TO WATERCROSS :BOATSPEED :STREAM-
SPEED
       MAKE "POST [ 130 -50 ]
       RIVER
       CROSS
       (PRINT [ DOWNSTREAM DISTANCE IS
] (YCOR + 50))
       (PRINT  [  FOR  BOAT SPEED OF ]
:BOATSPEED)
       (PRINT [  AND STREAM SPEED OF ]
:STREAMSPEED)
       END


    TO RIVER
       HT CS PU BK 50 RT 90 PD
       REPEAT 2 [ BANK ]
       SETPOS [ -130 -50 ]
       SETHEADING TOWARDS :POST
       ST
       END


    TO BANK
       FD 130 LT 90 FD 240
       LT 90 FD 130
       END


    TO CROSS
       FD :BOATSPEED
       SETHEADING 0 FD :STREAMSPEED
       SETHEADING TOWARDS :POST
       IF (130 - XCOR < 2)  [ STOP ]
       CROSS
       END
```

If you work with WATERCROSS and wish to share your findings with others, please feel free to write up a brief report or note and send it to us. We would be interested especially in knowing about your extensions or other applications of the procedures or the concepts.

Many times, there is a difference between the "perfect theoretical solution" to a problem and the actual solution which works in the real world. WATERCROSS is one small example of how LOGO can help us to explore differences such as this quickly and in an interesting way.

FORWARD 100!

Tipps for Teachers continued

Apple LOGO manual which randomly positions a target box on the screen and randomly sets the location and direction of the turtle. I might also use this TARGET procedure for early learners of LOGO, but I think its most appropriate use would be as a later project for students to create without the manual.

By the time the learners have explored and mastered distance and direction, they are probably ready to describe the size of the turtle screen across the top, bottom, and the middle, and along both sides and the diagonal. They should be able to guess distances fairly well in turtle steps. The understanding of turtle turns and their size generally lags behind the understanding of distance. I do not teach "degrees" as a prerequisite description of direction ... however, I do not hide the term, either. "Turtle Turns" does very well for younger children. "Square corner" is as explanatory as "right angle" or "90 degrees," although choice of words changes with the learner. A great advantage of LOGO is that prior knowledge of terminology does not limit even the youngest learner from creating. However, incomplete or inaccurate concrete understanding of concepts of distance and direction will inhibit the more sophisticated "turtling." Without adequate preparation, the power of programming with LOGO will be diminished.

Steve Tipps is a professor at the University of Virginia, and has presented LOGO workshops for teachers throughout the eastern United States.

SPIN continued

For the TI 99/4 and 99/4A:

```
     TO SPIN
        MAKE "HUNDREDS RANDOM
        TEST :HUNDREDS<4
        IFF SPIN
        MAKE "TENS RANDOM
        TEST :HUNDREDS=3
        IFT TEST :TENS>5
        IFT SPIN
        MAKE "ONES RANDOM
        MAKE "DEGREES (:HUNDREDS *
100) + (:TENS * 10) + :ONES
        SETHEADING :DEGREES
        END
```

An additional application of the SPIN procedure might be in scouting activities. As mentioned in Linda Nix's page one article in this issue, many of her students made rapid LOGO progress as a result of previous scouting experience with compasses. The SPIN procedure could be used by scouts to practice estimating headings by observing the orientation of the turtle on the screen. Their guesses could be confirmed by PRINT :HEADING.

continued

Microworlds continued

which can resursively call themselves is the result of memory and speed limitations, rather than a conscious decision that a language would be improved by the absence of this capability. Therefore, if these features are desirable, a continuing review is necessary to ensure that the omission of the feature does not continue after the technologic constraints are removed.

LOGO was developed over a number of years in an environment in which cost was not a constraining factor. The only stated limitation was usability of a characteristic in a learning environment. This process was begun over a decade ago. Now some of the benefits of this effort are being reaped as many of these characteristics are becoming possible in microcomputer-based implementations.

Three important aspects of LOGO are embodied in the fact that it is interactive, procedural, and extensible. This by no means exhausts the list of capabilities which define LOGO. Other aspects such as encouragement of meaningful procedure and variable names by removing restrictions on their length, the presence of a resident screen editor, and effective means of passing parameters between procedures all contribute to the underlying quality of LOGO.

Some components of LOGO can be altered or deleted without seriously degrading its fundamental structure. Some implementations of LOGO have arrays, sprites, and floating-point arithmetic, while others do not. Depending on the intended use, these omissions may not seriously cripple the language. However, beyond a point it becomes questionable whether a language can seriously be described as "LOGO," given the original intent of the language. Effective utilization of LOGO will ultimately depend on teachers who understand its characteristics, and the reasons for their inclusion in the language.

Glen Bull is a professor at the University of Virginia, and has taught LOGO courses at the graduate and undergraduate level.

Also, if you are observant, you will notice that the SPIN procedure on the TI computers will orient the turtle to certain "temporary" headings on occasion. The explanation for this is dependent upon a full understanding of the LOGO iterative process. Can you figure it out?

FORWARD 100!