

## **PRE-SERVICE ELEMENTARY TEACHERS MAKE CONNECTIONS BETWEEN GEOMETRY AND ALGEBRA THROUGH THE USE OF TECHNOLOGY**

Doris J. Mohr  
University of Southern Indiana  
Department of Mathematics  
Evansville, IN 47712  
[djmohr@usi.edu](mailto:djmohr@usi.edu)

### **Abstract**

*In a geometry content course for pre-service elementary teachers, technology was utilized to assist students in making sense of shapes. They learned to write simple procedures in Logo that would program a turtle to draw various quadrilaterals. In the context of writing these procedures, the pre-service teachers used variables to represent the lengths of sides and the measures of angles. As a result of these activities, students' conceptual understanding of variables changed.*

In 2000, NCTM (National Council of Teachers of Mathematics) released its *Principles and Standards for School Mathematics* (NCTM, 2000). The five process standards – problem solving, reasoning and proof, communication, connections, and representation – address effective methods by which students should learn mathematics. The use of technology is endorsed as a way to engage students in solving problems not possible by more traditional means, to engage students in reasoning, and to make connections between alternative representations.

There are basically three paths teachers can follow when using computers in the classroom (Clements, 2000). They can use computer games mainly for reward purposes; they can integrate drill exercises and other structured computer tasks; or they can use problem-solving software to engage students in higher-order skills. According to Clements, “Research suggests that the first path leads nowhere educationally useful.... The second path is educationally plausible....The third path is more challenging – in time, in effort, in commitment, and in vision. This path alone, however, offers the potential for substantive educational innovation consonant with NCTM Standards” (p. 42).

Mousley, Lambdin and Koc (2003) recommend that mathematics teacher educators use technology in the same way they expect their future teachers to use technology with children. Unfortunately, many mathematics educators have not had experience teaching with technology. In a recent study (Kersaint, Horton, Stohl, & Garofalo, 2003) of beliefs and practices of mathematics teacher educators (MTEs), it was found that MTEs are not, in general, comfortable with integrating technology into classes for pre-service teachers. They lack the experiences with using technology as well as access to the most recent recommendations for implementing it.

The purpose of this paper is to share a series of Logo activities that were developed and used with pre-service elementary teachers in a one-semester geometry course. Logo is a computer programming language designed particularly for learning. In most versions of Logo, the image of a turtle moves across the screen in response to a set of commands (see Turtle Pond at <http://illuminations.nctm.org>). The online version of the Principles and Standards for School

Mathematics contains electronic examples that illustrate many of the standards. One of these (4.3 Learning Geometry and Measurement Concepts by Creating Paths and Navigating Mazes) utilizes a ladybug instead of a turtle to draw paths and rectangles and to navigate through mazes (<http://standards.nctm.org/document/eexamples/index.htm>). My university has a site license for Logo, but there are also some versions that can be downloaded for free (see, for example, <http://www.softronix.com>). In developing these activities, I relied on *Lessons in Logo: Logo Works* (Cory & Walker, 2000) for ideas. The main purpose of these activities was to help the pre-service teachers understand shapes, their properties, and their classification. What I discovered, however, was that pre-service teachers also acquired a new understanding of variables.

Understanding variables is often confusing for students of all ages. According to Philipp (1992), the use of letters of the alphabet for mathematical purposes can be confusing to students because of the multiple roles they play. Letters can be used as labels, such as  $d$  to indicate distance; they can be used as an unknown in an equation, such as  $5x - 9 = 91$ ; they can be used for a constant, such as  $\pi$ ; they can be used as generalized numbers, such as in  $a + b = b + a$ ; they can be used as quantities that vary, as in  $y = 9x - 2$ ; they can be used as parameters, such as in  $y = mx + b$ ; or they can be used as abstract symbols, such as the identity  $e$  in abstract algebra (p. 560). In Logo, a variable takes on the role of a quantity that can vary. As one student wrote, “I didn’t realize a variable could stand for more than one number. I just thought it was an unknown.”

Unlike algebraic variables, a variable in Logo is not typically identified with one letter of the alphabet; instead, a word or phrase is used to identify the variable. And variables are used in nontraditional ways, such as to vary the length of a side or an interior angle of a polygon. As one student commented, “I have been forced to think of variables replacing numbers differently than I normally use them.” Still another commented, “I feel it showed real examples of how a variable can work.” Logo can thus provide students with alternative views of variables and provide a connection between algebra and geometry. In the remainder of this paper, I describe the activities the pre-service elementary teachers (PSETs) completed using Logo and how it changed their thinking about variables.

### Logo Activities

Students began by learning a few of the basic Logo commands. In response to the command *FD 100*, the turtle will move forward on the screen 100 turtle steps. Similarly, a command of *BK 100* moves the turtle back 100 turtle steps. To turn the turtle to its right or left, one can simply enter *RT 90*, *LT 45*, or a similar command that tells the turtle which direction to turn and how far. After trying out a few commands, the students soon realized that the turning angle is the *exterior* angle of a polygon, not the *interior* angle. For example, if a turtle is drawing a  $135^\circ$  angle, as shown in Figure 1, the turtle must turn  $45^\circ$  to its right. (Note: The dotted line has been inserted here to show the path the turtle will follow if no turn is made. This also helped the students determine the turning angle of the turtle.)

Students were then asked to write the commands to draw a square, a rectangle, a rhombus, and a parallelogram. These shapes had already been introduced in class, so they knew many of the characteristics of each shape. They were asked to write the commands on paper before they entered them into the computer to encourage them to use reasoning skills and not rely solely on

the visual image to determine their next command. If the turtle did not draw the desired shape, they were asked to re-evaluate the set of commands and try again. For example, to draw a rectangle, the following list of commands would be entered, resulting in the shape seen in Figure 2: *FD 50 RT 90 FD 70 RT 90 FD 50 RT 90 FD 70 RT 90*

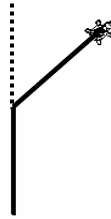


Figure 1. Illustration of turning angle.

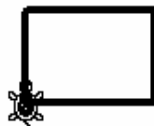


Figure 2. Rectangle drawn with Logo commands

The students were instructed to include the last command of *RT 90* to reposition the turtle in its starting position, which is in the middle of the screen, facing up. (The reason for including this final command soon became clear for the students.)

Next, the *REPEAT* command was introduced as a means to reduce the number of commands that need to be entered. For example, to draw the same rectangle using the *REPEAT* command, one can enter: *REPEAT 2[FD 50 RT 90 FD 70 RT 90]*. The turtle repeats the set of commands within the brackets the number of times indicated immediately following the word *REPEAT*. If the final command of *RT 90* is excluded in the original set of commands, students often make the mistake of using the repeat command incorrectly: *REPEAT 2[FD 50 RT 90 FD 70]* and the drawing in Figure 3 results.



Figure 3. Final *RT 90* omitted in *REPEAT* command

The next step was to introduce the students to writing procedures, which is a set of commands that can be executed by entering the name of the procedure. The use of procedures eliminates the task of typing the entire set of commands each time you wish to draw a particular shape. The students began with the task of writing a procedure that would draw a square whose

sides had length 50. They already knew that the commands for drawing such a shape were *REPEAT 4[FD 50 RT 90]*. To make this into a procedure, the REPEAT command needs to be put into the following format:

```
TO SQUARE50  
  REPEAT 4[FD 50 RT 90]  
END
```

Every procedure starts with the word TO and stops with the word END. The name of the procedure SQUARE50 (no spaces between letters) was specifically chosen to identify the type of shape and its size.

The procedure for a parallelogram proved to be the most challenging, for the students needed to concern themselves with the size of the angles, as well as the lengths of the sides. And they needed to recall that consecutive angles are supplementary. For example, the following procedure will draw a parallelogram with sides of length 50 and 70, and angles of  $45^\circ$  and  $135^\circ$  (see Figure 4).

```
TO PARALLELOGRAM  
  REPEAT 2[FD 50 RT 45 FD 70 RT 135]  
END
```



Figure 4. Parallelogram drawn using procedure

After the students became familiar with the format for writing procedures, variables were introduced. I explained to the students that variables would allow them to write more efficient procedures, ones that will draw a particular shape of any size. They would no longer need to write one procedure for a square whose sides had length 30, one for a square whose sides had length 40, and so on. The students began by examining procedures written by fictitious students. For example, suppose Jane wrote this procedure to draw a square:

```
TO SQUARE40  
  REPEAT 4[FD 40 RT 90]  
END
```

And suppose Fred wrote this procedure to draw a square:

```
TO SQUARE30  
  REPEAT 4[FD 30 RT 90]  
END
```

And Marge wrote this procedure:

```
TO SQUARE50
  REPEAT 4[FD 50 RT 90]
END
```

I then asked the students this series of questions:

Looking at the command line: *REPEAT 4[FD \_\_\_\_\_ RT \_\_\_\_\_ ]*, in all three sample procedures,

- 1) Which number, besides the 4, is the same in all the procedures?
- 2) What does this number represent in the square?
- 3) Which number changes or varies from one procedure to the next?
- 4) What does this number represent in the square?

After realizing that it was the length of the side that varied, or changed, a name was chosen for the variable. It is best to use a name that clearly identifies the quantity or property of the shape that changes. For example, words such as LENGTH, WIDTH, or SIDE could be used for this particular variable. To illustrate the differences and similarities between two procedures, one with a variable and one without, the students were shown the following two procedures. The one on the left will draw only a square whose sides have length 30, while the one on the right will draw squares of all sizes. The important differences are emphasized in bold.

```
To SQUARE30
  REPEAT 4[FD 30 RT 90]
END
```

```
To SQUARE :SIDE
  REPEAT 4[FD :SIDE RT 90]
END
```

In the first line, the name of the procedure is changed from SQUARE30 to simply SQUARE. The name of the variable must also appear on the first line, preceded by a pair of dots or colon. This pair of dots tells the turtle that the word that follows is a variable and not the name of a procedure. Inside the brackets, following the repeat command, the number 30 is replaced with the name of the variable.

When ready to execute, or run the program, an input is required that tells the turtle what number to substitute into the variable **SIDE**. For example, to tell the turtle to draw a square whose sides have length 50, one would type: SQUARE 50. To tell the turtle to draw a square whose sides have length 80, one would type: SQUARE 80, and so on. (When executing the procedure, a space is required between the name of the procedure and the input for the variable.)

After this introduction to variables, the students were assigned the task of writing a procedure that would draw any size rectangle. This was more challenging because two variables were needed, one for the length and one for the width. One such procedure is:

```
To RECTANGLE :WIDTH :LENGTH
  REPEAT 2[FD :WIDTH RT 90 FD :LENGTH RT 90]
END
```

To execute this procedure, one would type `RECTANGLE 50 70`, for example, producing a rectangle that has a width of 50 and a length of 70, as shown in Figure 5.

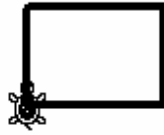


Figure 5. Rectangle drawn using procedure with variables.

If the students executed the procedure by typing `RECTANGLE 70 50`, reversing the order of the inputs, the rectangle in Figure 6 resulted. The first number after the word `RECTANGLE` will be put into the first variable listed on the top line of the procedure. The order of inputs did not, in this case, affect the ultimate outcome; just the orientation of the rectangle changed. The order of inputs became more important when they tried to execute the procedure for a parallelogram.

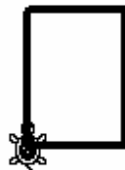


Figure 6. Rectangle drawn with inputs reversed.

Writing the procedure to draw a parallelogram was particularly challenging for many of the students because they had to define the variable for the second angle in terms of the first. For example, if the variable chosen for the size of the first angle was `ANGLE`, then the variable for the second angle could be defined as `180 - :ANGLE`. The students often wanted to use two different variables, such as `ANGLE1` and `ANGLE2`, until they remembered that the two measurements were related. Of course the procedure would work with two different variables for the angles, but students were encouraged to use the minimum number of variables required. In this way, they had to focus on the properties of the shape. Here's one example of a parallelogram procedure:

```
To PARALLELOGRAM :SIDE1 :SIDE2 :ANGLE
  REPEAT 2[FD :SIDE1 RT :ANGLE FD :SIDE2 RT 180 - :ANGLE]
END
```

To execute the procedure, the name of the procedure is typed, followed by three inputs. The three inputs must correspond with the three variables listed in the first line of the procedure and in the same order. The first input, therefore, must give the length of one of the sides, the second input must give the length of an adjacent side, and the third input must give the size of one of the angles. For example, if one were to type `PARALLELOGRAM 50 100 45`, the shape on the left in

Figure 7 results. If the lengths of the two sides are interchanged, the shape on the right in Figure 7 results. Again, the same parallelogram still results, just the orientation is different.

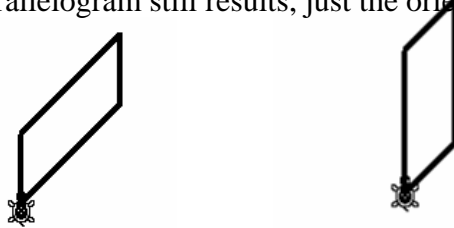


Figure 7. Parallelograms drawn when inputs for lengths of sides are interchanged.

However, if the input for the angle is interchanged with the input for one of the sides, such as in PARALLELOGRAM 45 50 100, the shape in Figure 8 results.

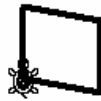


Figure 8. Parallelogram drawn when angle input becomes side input.

Even more interesting anomalies resulted when the students tried to draw a rectangle with the parallelogram procedure. If, for example, they wanted the two sides of the rectangle to be 50 and 100, they should type: PARALLELOGRAM 50 100 90. The last input of 90 gives the parallelogram four right angles. But if they typed: PARALLELOGRAM 90 50 100 instead, they did not get a rectangle. The correct output is shown on the left in Figure 9, and the incorrect output on the right. Because pre-service elementary teachers have probably not seen functions of two variables for some time, such as  $f(x,y) = 2xy + 3x^2$ , they did not remember that order matters:  $f(2,3) \neq f(3,2)$ . As one student wrote, “I’ve realized that a variable can be anything and if you substitute the wrong thing into the equation it can mess up everything.”



Figure 9.

Correct order of inputs results in figure on the left. Incorrect order of inputs results in figure on the right.

The final activity the students did involved writing procedures to draw two regular polygons – an equilateral triangle and a regular hexagon. Once again, they were asked to use a variable to ensure that the turtle could draw different-sized polygons. These two procedures were fairly simple for them to complete, after writing the parallelogram procedure. This proved to be more challenging and many of them needed help in determining which attributes of the regular polygon they wanted to vary. The necessity of changing the length of each side was obvious immediately. But most thought they needed two more variables – one for the number of sides and one for the size of the exterior, or turning, angle. After being reminded that the size of the

exterior angle is dependent upon the number of sides, they realized that one more variable would suffice. Here is one procedure for drawing regular n-gons:

```
TO REGULARPOLYGON :LENGTHOF SIDE :NUMBEROF SIDES  
  REPEAT :NUMBEROF SIDES[FD :LENGTHOF SIDE RT 360 / :NUMBEROF SIDES]  
END
```

Once the procedure is written, many interesting designs can be made. One such design is shown in Figure 10 and is the result of entering the sequence of commands below. (Note: The turtle has been hidden using command HT.) An interesting follow-up question for this activity is to have students determine the commands that will draw a circle.

```
REGULARPOLYGON 40 3  
REGULARPOLYGON 40 4  
REGULARPOLYGON 40 5  
REGULARPOLYGON 40 6  
REGULARPOLYGON 40 7  
REGULARPOLYGON 40 8  
REGULARPOLYGON 40 9  
REGULARPOLYGON 40 10
```

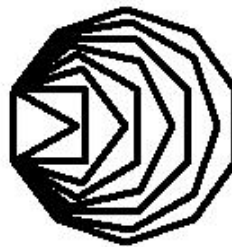


Figure 10. Shape drawn using regular polygon procedure repeatedly.

### **Impact on students' perception of variables**

Prior to the activity in which the pre-service elementary teachers used variables to write procedures, they were asked to complete the statement: *A variable is...* Approximately 77% of the sixty-nine students mentioned that a variable is an unknown, or a number to be solved for in an equation. Only 14.5% referred to a variable as a quantity that can change, or take on more than one value. After using variables to write procedures, only 49% of the pre-service elementary



teachers referred to a variable as an unknown or a number to be solved for in an equation, while 26% mentioned that variables can take on different values. That is almost twice as many as prior to the use of Logo. As one student wrote, “A variable is something that is changeable. For example, all hexagons are not the same size.”

After students had completed all of the Logo activities they were asked to respond to this prompt: *You’ve been working with variables this semester as you wrote procedures in Logo. Do you think working with Logo has made a difference in how you think about variables? If so, how?* Thirteen percent of the students indicated they either had a better understanding of variables or they had more confidence in using them, while 10% stated that they realized that variables have different uses. As one student wrote, “Working with Logo has helped me to understand more about variables and how to be more efficient when working with variables.” Another 9% of the students stated that Logo helped them to visualize the effects of variables. One student summed it up by stating, “Yes, it made it a little bit easier to understand because I had a picture to look at after I did the command.”

These two prompts were aimed directly at what students thought about variables. For me, the more revealing data came when I analyzed the reflections that students completed after each Logo activity. I asked them to respond to the written prompt: *What did you learn from completing this activity -- About yourself as a problem solver? About geometry?* Despite the fact that variables were not specifically referred to in the question, twenty-two percent of the students’ responses contained comments about variables. Nine percent of the students mentioned that they understood variables better and/or mentioned that their confidence in using variables had increased. Another nine percent mentioned that variables have many different uses. The fact that they mentioned variables when prompted to explain what they learned about geometry told me that Logo did, indeed, have an impact on their understanding of and attitudes toward variables.

### **Summary**

These particular Logo activities proved to be an effective way to help my pre-service elementary teachers make connections between geometry and algebra and to broaden their understanding of variables. By incorporating individual written explanations and reflections, I was able to incorporate all five process standards – problem solving, reasoning and proof, communication, connections, and representation – into the lessons. The higher-order thinking skills required to complete these activities was, I believe, the type that “offers the potential for substantive educational innovation consonant with NCTM Standards” (Clements, 2000, p. 42).

One benefit of using Logo, or any technology, with pre-service elementary teachers is that there is evidence to suggest that those who learn to use technology in the context of learning mathematics are “more likely to see its potential benefits and use it in their subsequent teaching” (Garofalo, Timmerman, & Shockey, 2000, p. 68). Although I will never know the full impact on these future teachers, my hope is that they will teach as they have been taught, using technology to engage their students in higher-order skills.

Seymour Papert, who along with a team from Bolt, Beranek and Newman led by Wallace Feurzeig, developed Logo in 1967 (Logo Foundation, 2000). His philosophy may still be relevant today:

In many schools today, the phrase ‘computer-aided instruction’ means making the computer teach the child. One might say the *computer is being used to program* the child. In my vision, *the child programs the computer* and, in doing so, both acquires a sense of mastery over a piece of the most modern and powerful technology and establishes an intimate contact with some of the deepest ideas from science, from mathematics, and from the art of intellectual model building. (Papert, 1980, p. 5)

### References

- Clements, D. H. (2000). From exercises and tasks to problems and projects: Unique contributions of computers to innovative mathematics education. *Journal of Mathematical Behavior*, 19, 9-47.
- Cory, S., & Walker, M. (2000). *Lessons in Logo: Logo works*. Cambridge, MA: Terrapin Software.
- Garofalo, J., Timmerman, M., & Shockey, T. (2000). Promoting appropriate uses of technology in mathematics teacher preparation. *Contemporary Issues in Technology and Teacher Education*, 1(1), 66-88.
- Kersaint, G., Horton, B., Stohl, H., & Garofalo, J. (2003). Technology beliefs and practices of mathematics education faculty. *Journal of Technology and Teacher Education*, 11(4), 549-577.
- Logo Foundation. (2000). What is Logo? Retrieved May 2, 2004, from <http://el.media.mit.edu/logo-foundation/logo/index.html>
- Mousley, J., Lambdin, D., & Koc, Y. (2003). Mathematics teacher education and technology. In A. J. Bishop, M. A. Clements, C. Keitel, J. Kilpatrick & F. K. S. Leung (Eds.), *Second international handbook of mathematics education: Part one* (Vol. 10, 395-432). Dordrecht: Kluwer Academic Publishers.
- National Council of Teachers of Mathematics. (2000). *Principles and standards for school mathematics*. Reston, VA: Author.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books, Inc.
- Philipp, R. (1992). The many uses of algebraic variables. *Mathematics Teacher*, 85(7), 557-561.