Using Spreadsheets to Enhance Understanding of Number Theory

Mario Toussaint, Don Ploger

Florida Atlantic University
3200 College Avenue, Davie, Florida 33314
E-mail: MTOUSSA1@fau.edu

Abstract

Computer spreadsheets can help elementary school students explore concepts in number theory. We describe a spreadsheet program that can generate all the factors of an integer. To understand how the spreadsheet solves these problems, we use the metaphor of a robot. The robot must interpret data from the real world and respond effectively. Although non-engineers may not understand the details, they can see what the robot types, and can discuss how the robot makes decisions.

Students can see mathematical knowledge being used. The robot can add, subtract, multiply, and divide, and determine whether a number is an integer. Based upon this knowledge, the robot can determine the factors of a number. In one method, the robot follows the rules blindly, testing each possible factor. In the second method, the robot uses knowledge of number theory to solve the problem much more efficiently.

The activities are extended to include the topic of prime numbers. In the first method, the robot determines that 97 is prime by performing all possible divisions starting with 1. Although the answer is correct, the method is inefficient. It is much more effective to apply knowledge of number theory to determine that only the prime numbers less than ten need to be tested. As a result, only four divisions, rather than 97, are needed to determine the correct answer. With the power of spreadsheets, students can observe different methods that get the correct answer, and discover those that are most efficient.

1. Introduction

This research is guided by a constructivist perspective: knowledge flows from prior knowledge and it emerges in activity (Dewey, 1938; Bruner, 1977; Kolb, 1976). In this particular case, we are teaching about number theory by building upon a student's existing knowledge of multiplication and division. We follow Piaget's general guidelines of ensuring that instruction is appropriate to students' level of cognitive development (Piaget, 1972). We are also guided by Papert's constructionist approach in which students build their own models using computer programs (Papert, 1980). There are, however, important differences from Papert's approach. Papert focused on a particular form of geometry that differed from the standard K-12 approach. Although there are some advantages to using exterior angles--the sum of all exterior angles of any polygon equals 360--this is so different from the standard approach, that it proved difficult to bring the working two schools. Furthermore, Papert used a particular programming language that did not correspond to classroom practice.

In this study, we focus on Number theory, a topic well-established for grades four and five. In fact, this topic is essential in both common core as well as prominent Home school approaches. Furthermore, instead of creating a new computer language, this study uses spreadsheets which are readily available in most public schools.

Spreadsheets can display numbers and manage arithmetic so that students are free to observe mathematical patterns. The lessons use the metaphor of a robot moving through the spreadsheet performing arithmetic, making decisions, and writing the results. The robot interprets data from the real world and responds effectively.

In some ways, this is similar to the robot from the original Logo Turtle Geometry. Logo was highly influential on educational research in the 1980s (Papert, 1981). In the mid 1960s Seymour Papert, a mathematician who had been working with Jean Piaget in Geneva, came to the United States where he co-founded the MIT Artificial Intelligence Laboratory with Marvin Minsky. The Logo Programming Language, a dialect of Lisp, was designed as a tool for learning. Its features — modularity, extensibility, interactivity, and flexibility — follow from this goal. Turtle geometry had a strong influence on educational research (Abelson & diSessa, 1981). Although Logo remains respected (Rowe, 2007), it had hardly any influence on the schools. There are two reasons for this failure. First, it requires students and teachers to learn some programming in a language that is not freely available. Second, it teaches geometry to elementary school students in a very different way than in the standard K-12 curriculum.

In this study, we use computer spreadsheets, which are almost universally available. Furthermore, we emphasize a topic that is central to the elementary school curriculum: factors and prime numbers. Students imagine a robot moving through the spreadsheet performing arithmetic, making decisions, and writing the results. The robot must interpret data from the real world and respond effectively.

Although non-engineers may not understand the details, they can see what the robot types, and can discuss how the robot makes decisions. We use a computer spreadsheet as a tool, and a robot as a metaphor. We apply these engineering themes to elementary school education, specifically the topic of factors and prime numbers.

2. Spreadsheet Scenario: Factors of 30

The system can quickly provide all the factors of the given positive whole number using the division algorithm. The robot is programmed to systematically test all positive integers starting with one until it finds all factors of the number entered. For example, the student could attempt to find all the factors of 30.

In figure 1, the robot starts out by dividing 30 by 1 and places 1 directly below the number entered; it recognizes that 1 is a factor since the division yields a remainder of zero. Because 1 is factor of 30, the robot moves to the right and types 1 in the "low factor" column, and then moves one step further to the right and types 30 in the "high factor" column.

In figure 2, the robot proceeds to the next whole number and divides 30 by 2. It displays 2 under the 1 and places 2 in the "low factor" column as well as 15 in the "high factor" column. The robot was able to determine that 2 and 15 are also factors of 30 as 30 divided by 2 equals 15 and the remainder of the division is zero.

In figure 3, the robot displays the next number being tested and divides 30 by 3. It finds that 3 and 10 are factors of 30 since 30 divided by 3 equals 10 and the remainder of the division is zero. It continues to enumerate the factors of 30 by placing 3 in the "low factor" column and 10 in the "high factor" column.

In figure 4, the robot displays the next whole number tested and divides 30 by 4. It recognizes that 4 is not a factor of 30 since 30 divided by 4 yields a remainder of 2. It then moves to the right and displays that 4 is not a factor.

In figure 5, the robot tests the next whole number and calculates 30 divided by 5. It declares that 5 and 6 are factors of 30 as the division yields a remainder of zero. It places 5 in the "low factor" column and 6 in the "high factor" column.

Finally, in figure 6, the robot displays that the number 30 has no more factors. It calculated the square root of 30, kept the answer in its memory, and compare each factor found from each of the steps discussed above to the square root of 30. The robot knows that all factors were found when the next integer to be tested, in this case 6, is greater that the square root of 30.

Using the Robot for Discovery Learning

The Robot can perform simple computations such as add, subtract, multiply, divide two numbers, calculate the square root of a number, and compare two numbers; it can apply mathematical knowledge, such as determining whether a number is an integer, to make decisions based on the results of the calculations; and it can move along different paths according to some pre-determined criteria. In this case, the robot can move in three ways: (1) it can move directly down within the column to the next row; (2) it can move to the right, within the row, to the next column; and (3) it can move to the far-left column, within the row.

During this process, the robot compared every positive whole number starting with 1 and the square root of the given number, in this example 30, to determine whether all the factors were found. Students can discover this fact after using the robot to find all the factors of several different positive integers. This process will also teach the students the algorithm to generate all the factors of any given whole number. Teachers can use this process to help their students learn the concepts of prime and composite numbers through discovery. The instructor can prompt the students to find the factors of several prime numbers and help the students differentiate between prime and composite numbers. Further, teachers can ask students to formulate definitions for these concepts in their own words.

What does the robot know?

The robot can add, subtract, multiply, and divide. He can also tell whether a number is an integer. Based upon this knowledge, the robot can determine the factors of a number. It can follow these rules blindly by testing each number. In this case, the robot uses his arithmetic skills, but relatively little understanding of number theory.

The robot can learn to apply some mathematical knowledge. Factors occur in pairs. For example, 30 has eight factors arranged in four pairs: (1, 30), (2,15), (3, 10), and (5, 6). Since 6 appears as the larger number in a pair, the robot knows he has already found all the factors.

Furthermore, the robot can extend his mathematical knowledge. For the example shown, in each case, the smaller factor is less than the square root of 30. Therefore, in the case of 30, the robot only needs to test the numbers 1 through 5. By including the pairs of each of these factors (1, 2, 3, and 5), he knows he has found all the factors of 30.

The robot can also discover, for example, with 36, that he needs to test 6 as well. In this case, 6 is the square root of 36. Based upon this, the robot knows to test each number that is less than or equal to the square root of n. Because of these capabilities, the robot can show how to find the factors of any number up to 120 by testing only the numbers 1 through 10.

3. Scenarios:

Figure 7 shows how the robot can systematically test all positive whole numbers from one to ten to ascertain which, if any, of these numbers divide 10 with a remainder of zero. The robot starts with 1 and tests whether 10 divided by 1 yields a remainder of zero. The robot finds that the remainder is zero since 1 divides every whole number and displays that 1 is factor. The robot proceeds to the next whole number and checks whether 10 divided by 2 results in a zero remainder. Since the remainder of the division is zero, the robot outputs that 2 is a factor. The robot proceeds to examine the next whole and divides 10 by 3. The robot finds a non-zero remainder and concludes that 3 is not a factor of 10. The robot continues in this manner and tests 4, 5, 6, 7, 8, 9, and 10. It classifies 3, 4, 6, 7, 8, and 9 in the category of non-factors of 10, as the divisions did not yield a zero remainder and puts 1, 2, 5, and 10 in the group of factors of 10. It knows to stop at 10 since the factors of a whole number cannot be greater than the number itself.

Factors of 100

Figure 8 shows the factors of 100. Every number from 1 through 10 is tested, and the results indicate that 1, 2, 4, 5, and 10 are factors. Furthermore, since 100 ÷ 1 = 100, this means that 100 is also a factor. Following similar reasoning, 20, 25, and 50 are also factors.

It is clear that these numbers shown: 1, 2, 4, 5, 10. 20, 25, 50, and 100 are factors of 100. But how can we be certain that these are the only factors of 100? We did not test 11. Of course, we know that 100 is not evenly divisible by 11. But how did we know that before considering the example? How can we be certain that 12, 13, 14, and 15 cannot be factors of 100 without doing any more tests?

We know that 10 * 10 = 100. If a number, n, greater than 10 is a factor of 100, then 100/n must be less than 10. Therefore, we only need to test whether the numbers from 1 through 10 are factors of 100.

Divisibility tests

By definition, a prime number has exactly two factors, 1 and the number itself. A composite number has more than two factors. In the elementary school, it is essential that students find all the factors for each number up through 10, and it is useful to list the factors for numbers up to 25. With a list of the factors for a number, it is easy to tell whether it is prime or composite.

For larger numbers, a different method is needed. The strategy is to help students notice that for any composite number, the smallest factor (other than 1) is always prime. Therefore, we only need to test prime numbers. Furthermore, we only need to test numbers that are less than or equal to the square root of the given number. Therefore, for any number up to 100, we only need to test the prime numbers less than 10, namely 2, 3, 5, and 7.

Figure 9 shows the results of the “Is it prime?” example for 87. The XL program concludes that 87 is composite because there is an additional factor (namely, 3). The student could find that a display of all the factors of 87, gives the complete factor list (and there are a total of four factors, confirming that 87 is composite).

Figure 10 shows the results of the “Is it prime?” example for 97, and show that it is prime because all divisibility tests are false. We also need to show that there is no possible factor (other than 1 and 97). The complete factor list shows that there are exactly 2 factors, confirming that 97 is prime.

4. Summary

There are two distinctly different methods to determine whether a number is prime. In the first method, the robot determines that 97 is prime by performing all possible divisions starting with 1. Only two of those divisions yield whole number quotients. Although the answer is correct, the method is inefficient. It is much more effective to apply knowledge of number theory to determine that, for any number up to one hundred, only the prime numbers less than ten need to be tested. As a result, only four divisions, rather than 97, are needed to determine the correct answer. With the robot metaphor and the power of spreadsheets, students can observe different methods that get the correct answer, and discover those that are most efficient.

Acknowledgements

We thank Sandy Tang for helpful suggestions throughout the development of this paper.

References

[1] Abelson, H. & diSessa, A. Turtle Geometry (1981)

[2] Bruner, J. (1960). *The process of Education*. Cambridge, MA: Harvard University Press.

[3] Dewey, J. (1997). *Experience and Education*. New-York, NY:Touchstone.

[4] Kolb, D. (1976). *Learning style inventory technical manual*. Boston, MA: McBer.

[5] Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books.

[7] Piaget, J. (1972). *Intellectual Evolution from Adolescence to Adulthood*. Human Development.

[8] Rowe, (2007). FORWARD 40: WHAT BECAME OF THE LOGO PROGRAMMING LANGUAGE? Retrieved from https://www.wired.com/2007/10/forward-40-wher/

30	<- Enter a number < 121	
	Low Factor	Hi Factor
1	1	30

Figure 1

30	<- Enter a number < 121	
	Low Factor	Hi Factor
1	1	30
2	2	15

Figure 2

30	<- Enter a number < 121	
	Low Factor	Hi Factor
1	1	30
2	2	15
3	3	10

Figure 3

30	<- Enter a number < 121	
	Low Factor	Hi Factor
1	1	30
2	2	15
3	3	10
4	Not a factor	

Figure 4

30	<- Enter a number < 121	
	Low Factor	Hi Factor
1	1	30
2	2	15
3	3	10
4	Not a factor	
5	5	6

Figure 5

30	<- Enter a number < 121	
	Low Factor	Hi Factor
1	1	30
2	2	15
3	3	10
4	Not a factor	
5	5	6
6	No More Factors	

Figure 6

10	<- Enter a number < 121	
	Factor?	
1	Factor	
2	Factor	
3	Not a factor	
4	Not a factor	
5	Factor	
6	Not a factor	
7	Not a factor	
8	Not a factor	
9	Not a factor	
10	Factor	

Figure 7

100	<- Enter a number < 121	
	Low Factor	Hi Factor
1	1	100
2	2	50
3		
4	4	25
5	5	20
6		
7		
8		
9		
10	10	

Figure 8

87	<- Enter a number up to 100
Div Test	**Is a Factor?**
2	FALSE
3	TRUE: Smallest Prime Factor
5	FALSE
7	FALSE
87	is composite

Figure 9

97	<- Enter a number up to 100
Div Test	**Is a Factor?**
2	FALSE
3	FALSE
5	FALSE
7	FALSE
97	is prime

Figure 10