OzoBlockly Challenge: Fibonacci Sequence and Its Spiral

By Richard Born
Associate Professor Emeritus
Northern Illinois University
rborn2@niu.edu

One of the most fascinating sequences in the study of mathematics is the Fibonacci sequence, named after a 12th century Italian mathematician. The first two numbers in the sequence are both defined to be 1. Every number thereafter is the sum of the two preceding numbers. The first ten numbers in the sequence then become as shown in *Figure 1*. We see, in general, that for n > 2,

$$F_n = F_{n-1} + F_{n-2}. \quad \textit{(Equation 1)}$$

F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	…
1	1	2	3	5	8	13	21	34	55	…

Figure 1

Although there are many algorithms to compute the numbers in the Fibonacci sequence, in this challenge lesson we are going to use a technique known as recursion. This is by no means the most efficient way, but it will help you to learn how to use OzoBlockly functions and find out what recursion is. ***With recursion, a function calls itself***.

Equation 1 above for the Fibonacci sequence indicates that we could use recursion to compute successive numbers in the sequence. We can find the n^{th} Fibonacci number if we know the $(n\text{-}1)^{st}$ and $(n\text{-}2)^{nd}$ numbers in the sequence. *Figure 2* shows OzoBlockly code for a recursive function that can be used to compute Fibonacci numbers.

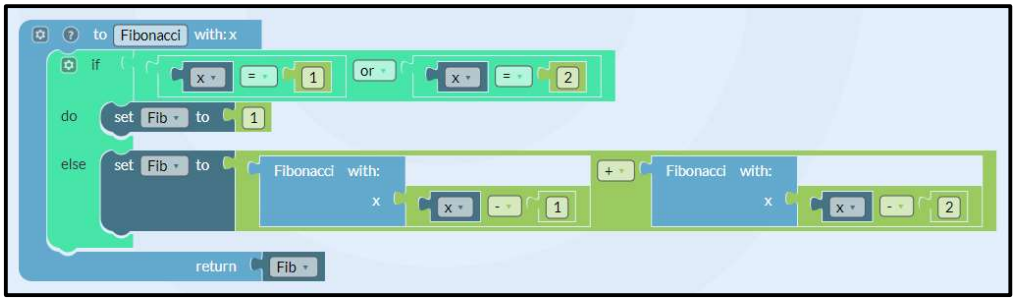


Figure 2

We have named the function **Fibonacci**. The function has an input we call **x**, which is the index number of the term that we wish to compute in the Fibonacci sequence. The function consists of an *if-then-else* statement. If x is 1 or x is 2, we simply set the return value *Fib* to 1, which are by definition the values of the first two terms in the sequence. Else, when x>2, we use recursion to call **Fibonacci**$_{x-1}$ and **Fibonacci**$_{x-2}$, which are added together as shown in Equation 1. Note that the OzoBlockly code "to Fibonacci with: x" is the ***function*** while the OzoBlockly code "Fibonacci with: x" is a statement that ***calls*** the **Fibonacci** function. You are encouraged to use the **Fibonacci** function of *Figure 2* in programming your solution to this challenge.

To get a feel for how the recursive **Fibonacci** function works, you can study the *recursive calling tree* for the sixth term F_6 in the sequence. This tree appears in *Figure 3*. Note that each of the leaves shown in green in the figure returns the value 1. With eight such leaves, the value of F_6 then turns out to be 8. There are five levels of recursion. You can imagine the inefficiency of this process as the value of *n* increases in computing F_n. Yet recursion does have a beautiful simplicity in its concept, and we can certainly program Evo using this recursive algorithm.

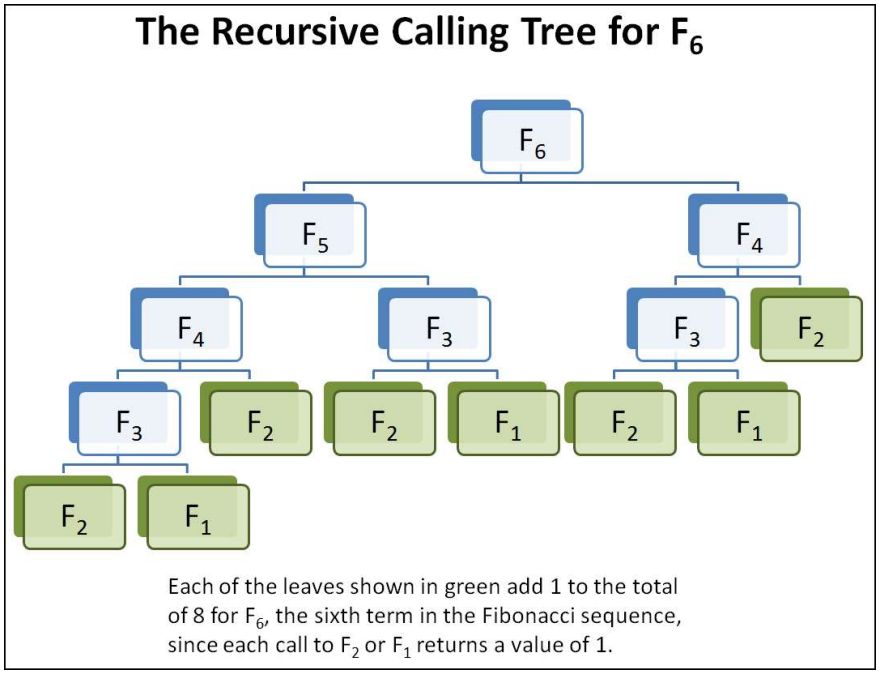


Figure 3

Figure 4 shows the maze that we will use to program Evo to follow the Fibonacci spiral. Note that the spiral starts out with two adjacent squares with sides of length 1. Adjacent to these squares is a square with sides of length 2. Adjacent to the previous squares is a square with sides of length 3. This process continues, with the sides of each new square having lengths equal to the terms of the Fibonacci sequence. Each of the squares is inscribed with a quarter circle whose radius is that of the corresponding Fibonacci sequence. The quarter circles, when

linked together, for the Fibonacci spiral. A larger copy of the maze, suitable for printing and using with Evo can be found on the last page of this document.

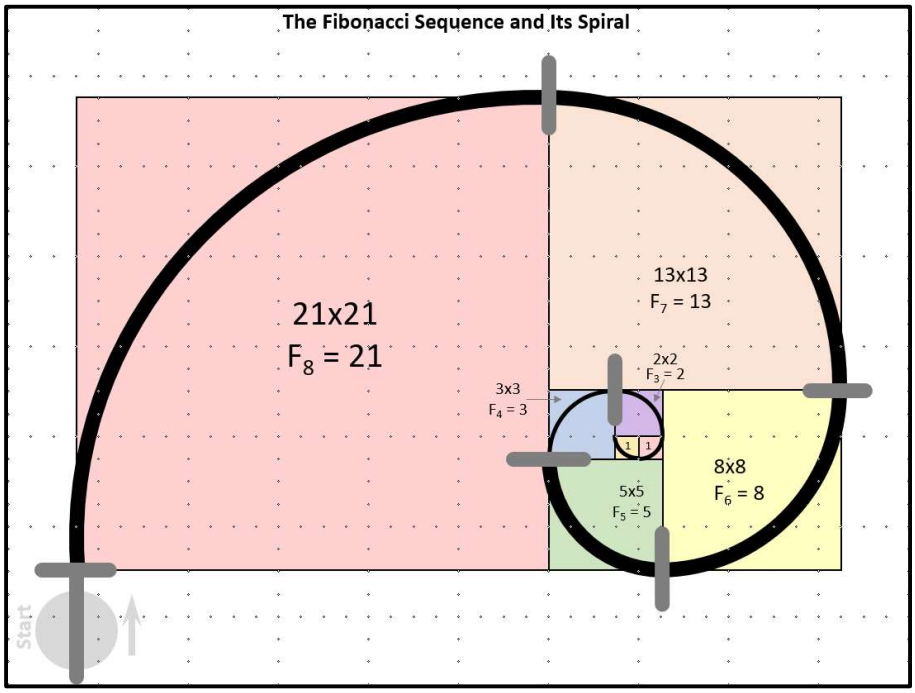
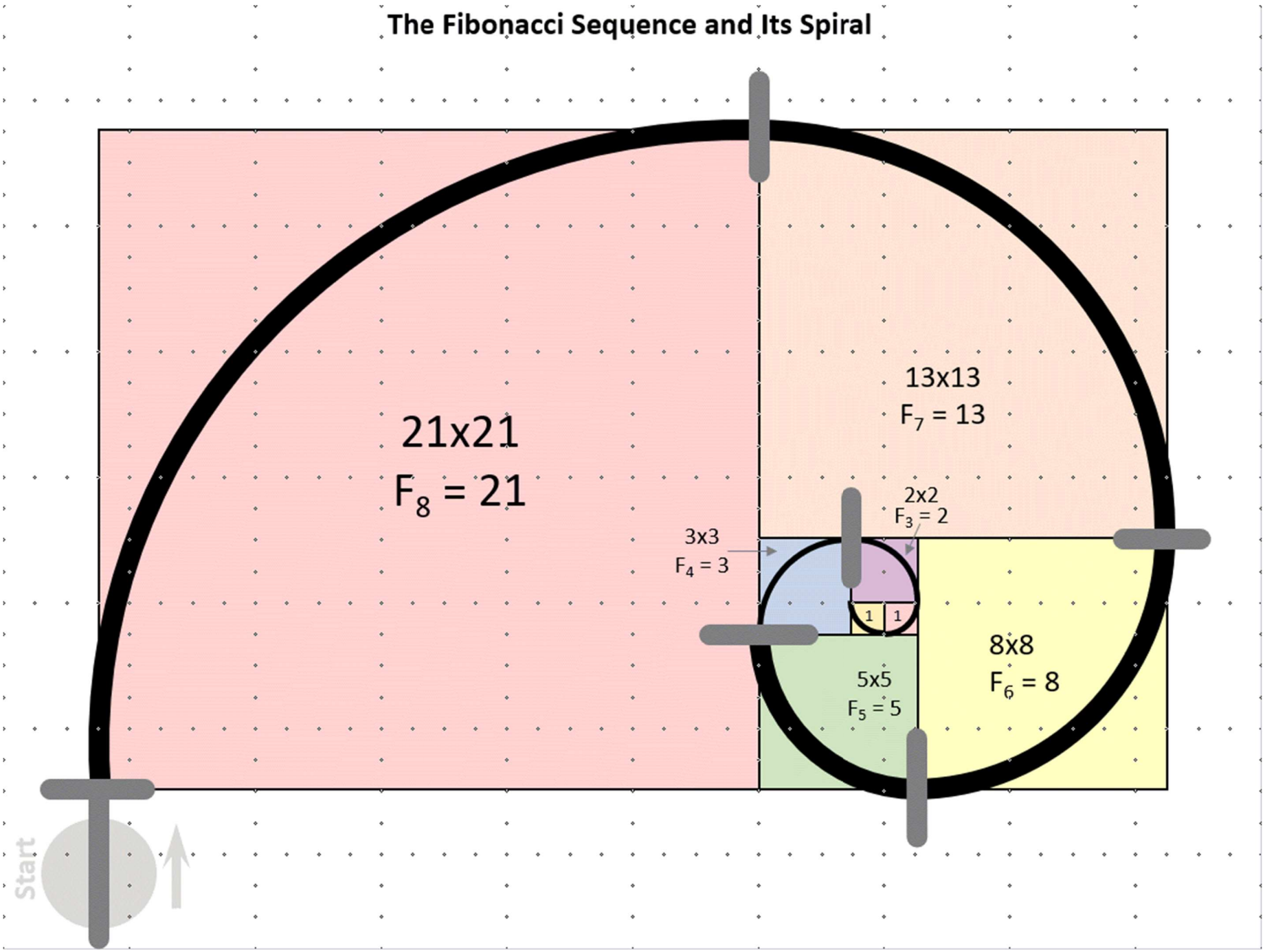


Figure 4

Challenge Requirements

1. Evo should begin at the start location in the lower left corner of the Ozomap facing the direction shown by the gray arrow.
2. As soon as he reaches the red 21x21 square, he should speak the value of F_8 (21) and then follow the arc in that square at a speed in mm of four times the value of the F_8 Fibonacci number.
3. As soon as he reaches the orange 13x13 square, he should speak the value of F_7 (13) and then follow the arc in that square at a speed in mm of four times the value of the F_7 Fibonacci number.
4. This process is to continue through the green 5x5 square.
5. The recursive algorithm for determining Fibonacci numbers must be used to determine the values that Evo speaks upon entering each of the squares.
6. The entire process must be done in a loop whose loop counter varies from 8 through 5 by steps of -1.

Have fun with this challenge lesson! You will have a great feeling of accomplishment when you have turned Evo into a perfect "Fibonacci Spiral traveler"!


The Fibonacci Sequence and Its Spiral
21x21
$F_8 = 21$
13x13
$F_7 = 13$
2x2
$F_3 = 2$
3x3
$F_4 = 3$
1
1
8x8
$F_6 = 8$
5x5
$F_5 = 5$
Start