Herramientas Computacionales para Ciencias
Homework 4

Mauricio Sevilla*

18/02/2019

Rules

This week we are going to concentrate on functions on `python`. The first point have to be saved on a file `problem1.txt`, and the rest on `problemN.py` where `N` symbolizes the number of the problem, the five files must be compressed on a file `userUniandes.zip` or `userUniandes.rar`, for example on my case it should be `j.sevillam.zip` or `j.sevillam.rar`.

Problem 1: Context [1/5]

- **0.25/1** What is pseudo-code?
- **0.25/1** What is a flow diagram?
- **0.25/1** What is a *passed by value* parameter on a function.
- **0.25/1** What is a *passed by reference* parameter on a function.

Problem 2: Factorial [1/5]

One of the most basic programs we can do by using recursive functions, is the factorial of a number!.

The factorial is defined as,

$$n! = n \cdot (n-1) \cdot (n-2) \cdots 1 \tag{1}$$

where by definition $0! = 1! = 1$.

Program a recursive function that calculates the factorial of a given number.

Problem 3: Fibonacci Sequence[1/5]

The Fibonacci Sequence also called the *golden ratio*, is one of the most known sequences, sometimes is represented as a spiral just as the following,

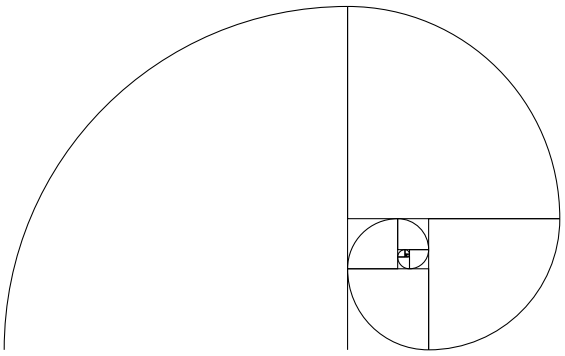

Figure 1: Fibonacci numbers plotted as the Golden Spiral

Those numbers can be easily calculated as follows

$$f(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ f(n-1) + f(n-2) & \text{if } n \geq 2 \end{cases} \tag{2}$$

*email=j.sevillam@uniandes.edu.co

Which means that the zeroth number is 0, the first is 1 and the following number can be calculated as the sum of the two previous ones, which means that the second should also be 1, the third is 2 and so on.

Write a code that, using a recursive calculate nth Fibonacci number.

Problem 4: Pascal's Triangle [1/5]

Pascal's triangle is a triangular array of the binomial coefficients. it is built as follows:

- You may start with the zeroth and first level, which are ones, and place them on a triangle

$$1$$

$$1 \quad 1$$

Figure 2: Initial Triangle

- Then, you add ones at the beginning and ending of the line, and the middle point is calculated by summing the two numbers above to the right and left, such that

$$1$$

$$1 \quad 1$$

$$1 \quad 2 \quad 1$$

Figure 3: Pascal's Triangle for $n = 2$

- Repeat the procedure such that, the following triangles are,

$$1$$

$$1 \quad 1$$

$$1 \quad 2 \quad 1$$

$$1 \quad 3 \quad 3 \quad 1$$

(a) $n = 3$

$$1$$

$$1 \quad 1$$

$$1 \quad 2 \quad 1$$

$$1 \quad 3 \quad 3 \quad 1$$

$$1 \quad 4 \quad 6 \quad 4 \quad 1$$

(b) $n = 4$

$$1$$

$$1 \quad 1$$

$$1 \quad 2 \quad 1$$

$$1 \quad 3 \quad 3 \quad 1$$

$$1 \quad 4 \quad 6 \quad 4 \quad 1$$

$$1 \quad 5 \quad 10 \quad 10 \quad 5 \quad 1$$

(c) $n = 5$

Figure 4: Pascal's Triangle for different value of n.

Implement a function that calculates the nth line of the Pascal's triangle.

Problem 5: Decode the Message [1/5]

Imagine that **Alice** and **Bob** have a message for you, but it is encrypted so that you have to write a program that decodes the message. The codification is very simple, you have to take the first letter from Alice's part and then one from Bob's, then the second of Alice's and then the second of Bob's, and so on. for instance, if Alice gives you `'hlo'` and Bob `'el.'` the mesage is `'Hello.'`.

Write a function that decodes the messages and try out the following message,

```
Alice=''Ti rga eoe esg o h ore"ermetsCmuainls'
 Bob='hspormdcdsamsaefrtecus Hraina optcoae"'
```