

Tutorials 1 - Recursion

Exercise 1

Write a recursive function `factorial(n)` that computes the factorial of n . Give the tree of recursive calls of `factorial(3)`.

Exercise 2

Write a recursive function `sequence(n)` that computes the term u_n of the following sequence :

$$\begin{cases} u_0 = 3, \\ u_n = 2 \times u_{n-1} + 1 \text{ if } n \geq 1. \end{cases}$$

This sequence is $(3, 7, 15, 31, \dots)$ and the call of `sequence(2)` returns 15.

Give the tree of recursive calls of `sequence(3)`.

Exercise 3

Write a recursive function `sequence(n)` that computes the term u_n of the following sequence :

$$\begin{cases} u_0 = 0, \\ u_1 = 1, \\ u_{n+2} = 2 \times u_{n+1} - u_n + 1 \text{ if } n \geq 0. \end{cases}$$

This sequence is $(0, 1, 3, 6, 10, \dots)$ and the call of `sequence(3)` returns 6.

Give the tree of recursive calls of `sequence(4)`.

Exercise 4

Write a recursive function `binomial(n,k)` that computes the binomial $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ using the following formula :

$$\begin{cases} \binom{n+1}{p+1} = \binom{n}{p} + \binom{n}{p+1} \text{ si } n > p \geq 0 \\ \binom{n}{0} = 1 \text{ if } n \geq 0, \\ \binom{n}{n} = 1 \text{ if } n \geq 0. \end{cases}$$

Exercise 5

Write a recursive function `search(T,e)` that takes as parameter an array T and an element e and that returns the first index of e in T .

For example, `search([1,1,3,4,4,5,6,8], 4)` returns the value 3.

Give the tree of recursive calls of `search([1,1,3,4,4,5,6,8], 4)`.

Exercise 6

Start to discover the turtle module of Python :

```
from turtle import *
forward(100)
right(120)
forward(50)
left(100)
forward(100)
```

What makes this program ?

Exercise 7

Use the primitives of the module *turtle* to write a program that draw the Koch snowflake.

The Koch snowflake is a drawing that can be obtained by following the following instructions :

- we start to draw the first image, that is a segment starting from position $[0, 0]$ and ending to position $[1, 0]$;
- From one image, we construct a new image by replacing each segment s by a drawing $f(s)$ defined in the following way :
 1. we cut the segment s in three that are equals ;
 2. we construct an equilateral triangle whose base are the segment in the middle ;
 3. we suppress the base of the new triangle.

You can consult the page of wikipedia for more informations.

Exercise 8

we call a composition of n a list of positive integers (≥ 0) such that the sum of all the integers are equals to n .

For example, all the compositions of 4 are $[1, 1, 1, 1]$, $[1, 1, 2]$, $[1, 2, 1]$, $[2, 1, 1]$, $[1, 3]$, $[3, 1]$, $[4]$.

Write a program that returns all the compositions of n .

Exercise 9

we call partition of n a list of decreasing positive integers such that the sum of all integers are equals to n .

For example, the partitions of 4 are $[1, 1, 1, 1]$, $[2, 1, 1]$, $[3, 1]$, $[4]$.

Write a program that returns all the partitions of n .

Exercise 10

We call permutation of n a list of n elements that contain all the integers from 1 to n .

For example, the permutations of 3 are $[1, 2, 3]$, $[1, 3, 2]$, $[2, 1, 3]$, $[2, 3, 1]$, $[3, 1, 2]$, $[3, 2, 1]$.

Write a program that returns all the permutations of n .