Pràctica de Recursivitat

TOTS ELS APARTATS VALEN EL MATEIX

1. **Fes un mètode iteratiu que calculi el factorial d’un número n. Bucle**

Exemple: El valor de 5! = 5 x 4 x 3 x 2 x 1.

Si voleu fer proves no poseu un n molt gran, que us pot consumir molta memòria.

1. **Fes un mètode recursiu que calculi el factorial d’un número n.** *Pista: intenteu separar la formula en dues parts.*
2. **Fes un mètode iteratiu al que li passarem una llista d’enters i ens retornarà la suma de tots els seus elements.**
3. **Fes que el mètode anterior  sigui recursiu**, de tal manera que calculi la suma del primer element de la llista mes la suma dels elements restants.

O sigui: **suma([a,b,c,d]) = a+suma([b,c,d])**

1. **Fes que el mètode anterior sigui recursiu**, però que vagi particionant la llista per la meitat, en dos subllistes, i calculi la suma de les 2 subllistes.

O sigui: **suma([a,b,c,d])=suma([a,b])+suma([c,d])**

1. **Fes un mètode recursiu que retorni el nombre de dígits de n.**
2. **Fes un mètode iteratiu que li doni la volta a una cadena de caràcters.** NOTA: Òbviament, si la cadena de caràcters és un palíndrom (ex: ada), la sortida de la funció serà la mateixa cadena inicial.

Exemple:

hola -> aloh

ssdafgt -> tgfadss

aabbddttabt -> aabbddttabt

ada -> ada

1. **Fes el mateix mètode però recursiu.**
2. **Programa els següents mètodes d’ordenació en Python:**

* Iteratiu: Bubble sort
* Recursiu: Merge sort

Compara el temps d’execució de cada mètode i compara’ls també amb el sort de Python. Utilitzeu la llibreria *time* per calcular el temps.

1. **Explica què resol la següent funció i posa-li un nom convenient:**

def misteri(x , y):

    if y == 0:

        return x

    else:

        return misteri(y, x % y)