

# TP3 – Fonction récursives

## Exercice 1 : Suite récurrente

Soit la suite définie par :  $\begin{cases} u_0 = 3 \\ u_{n+1} = u_n + \frac{3}{u_n} \text{ où } n \in \mathbb{N} \end{cases}$

1°) Ecrire une fonction *suitei*( $u_0, n$ ) itérative qui calcule  $u_n$ .

2°) Ecrire une fonction *suiter*( $u_0, n$ ) récursive qui calcule  $u_n$ .

## Exercice 2 : Algorithme d'Euclide

L'algorithme d'Euclide permet de calculer le pgcd de deux entiers naturels. Voici une fonction Python nommée Euclide qui implémente l'algorithme d'Euclide :

```
def euclide(a,b):
    while b!=0:
        a,b = b,a%b
    return a
euclide (66,11)
```

11

L'algorithme d'Euclide repose sur une idée simple : le PGCD de deux nombres a et b est le même que le PGCD de b et a % b (le reste de la division de a par b). Le processus continue jusqu'à ce que le reste soit zéro, et le dernier diviseur non nul est le PGCD.

- Réécrire cet algorithme à l'aide d'une fonction récursive *euclider*(a,b).

## Exercice 3 : Dichotomie

On considère une liste triée dans l'ordre croissant, par exemple :  $L = [5, 8, 13, 18, 20, 121]$ .

- Proposer une fonction récursive *dicho* qui admet comme arguments une liste triée L et un élément x. Cette fonction retourne True si x est dans la liste L, sinon elle retourne False. Elle utilise la méthode de dichotomie.

## Exercice 4 : Somme

1°) Ecrire une fonction récursive somme qui prend en paramètre une liste de nombre et renvoie la somme des termes de cette liste. On générera cette liste de façon aléatoire avec 10000 termes d'entiers compris entre 0 et 10.

2°) À l'aide du module time, on calculera le temps d'exécution de la fonction somme.

3°) Réécrire cette fonction de manière itérative à l'aide de la fonction intégrée « pop ».

## Exercice 5 : Palindrome

Un palindrome est un mot qui peut se lire dans les deux sens : KAYAK, LAVAL...

1°) Ecrire une fonction itérative qui répond « True » si le mot est bien est un palindrome et « False » dans le cas contraire.

2°) Ecrire une fonction récursive qui effectue la même tache.