

Python-ECG1-05-boucles-for-cor

October 5, 2021

1 TP5 - Boucles for - Corrigé

1.1 Exercice 1 - Calculer le terme de rang n d'une suite récurrente

Soit $(u_n)_{n \in \mathbb{N}}$ la suite définie par

$$\begin{cases} u_0 = 1 \\ \forall n \in \mathbb{N}, u_{n+1} = 2u_n - 3n \end{cases}$$

- 1) Sur feuille
- 2) Écrire une **fonction** python, appelée **suite_n(n)**, renvoyant la valeur de u_n . Ce programme et le tableau précédent doivent être cohérents.

```
In [1]: def suite_u(n):  
        u = 1  
        for k in range(n):  
            u = 2*u-3*k  
        return u
```

```
In [2]: #Tester ici la fonction  
        suite_u(10)
```

```
Out[2]: -2015
```

1.2 Exercice 2 - Les compteurs

On considère une urne contenant 5 boules rouges et 2 boules jaunes.

- 1) Comment simuler un tirage dans cette urne en utilisant **rd.randint(1,8)** ?

```
In [17]: import numpy.random as rd  
  
        r = rd.randint(1,8)  
  
        if r<=5 :  
            print('Rouge')
```

```

else :
    print('Jaune')

# On testera plusieurs fois.

```

Rouge

2) Simuler 100 tirages dans cette urne et compter le nombre de boules rouges obtenues.

```

In [22]: compteur = 0    #compteur du nombre de boules rouges

for k in range(100) :
    r = rd.randint(1,8)
    if r<=5 :
        compteur = compteur + 1

print(compteur)

```

81

1.3 Exercice 3 - Suite récurrente d'ordre 2

Soit $(u_n)_{n \in \mathbb{N}}$ la suite définie par

$$\begin{cases} u_0 = 1, u_1 = 2 \\ \forall n \in \mathbb{N}, u_{n+2} = 2u_n - u_{n+1} \end{cases}$$

Écrire une **fonction** python **Suite3(n)** qui, étant donné un entier naturel n , renvoie la valeur de u_n .

```

In [23]: def suite3(n):
    u = 0
    v = 2
    for k in range(n-1):
        w = 2*u-v
        u = v
        v = w
    return u

```

```

In [24]: # tester ici la fonction
suite3(10)

```

Out[24]: 342

1.4 Exercice 4

On appelle suite de Syracuse toute suite d'entiers naturels $(u_n)_{n \in \mathbb{N}}$ définie de la manière suivante : $u_0 \in \mathbb{N}^*$ et pour tout entier naturel n :

$$u_{n+1} = \begin{cases} \frac{u_n}{2} & \text{si } u_n \text{ est pair;} \\ 3u_n + 1 & \text{si } u_n \text{ est impair.} \end{cases}$$

Écrire une **fonction** python **Syracuse(u0,n)** qui, étant donné un nombre $u_0 \in \mathbb{N}^*$ et un entier naturel n , renvoie la valeur de u_n .

```
In [27]: def Syracuse(u0,n):
         u = u0
         for k in range(n):
             if u%2 == 0:
                 u = u/2
             else:
                 u = 3*u+1
         return u
```

2) Afficher les 20 premiers termes de la suite en partant de $u_0 = 15$.

```
In [28]: for k in range(20):
         print(Syracuse(15,k))
```

```
15
46
23.0
70.0
35.0
106.0
53.0
160.0
80.0
40.0
20.0
10.0
5.0
16.0
8.0
4.0
2.0
1.0
4.0
2.0
```