IA. 2 – Les boucles FOR - for … in range(…)

Exécution pas à pas de terme() :

5→u
1→k
$2 \times 5 - 4 = 6 \rightarrow$ u

$2 \rightarrow$ k
$2 \times 6 - 4 = 8 \rightarrow$ u
$3 \rightarrow$ k
$2 \times 8 - 4 = \mathbf{12} \rightarrow$ u *renvoyée*

```
def terme() :
    u=5
    for k in range(1,4) :
        u=2*u-4
    return u
```

Que fait terme() ? *Calcule et renvoie le 4ème terme de la suite $u_{n+1} = 2u_n - 4$ dont le 1er terme est 5*
Si on commence à u_0, il calcule u_3, si on commence à u_1, il calcule u_4)

Quelle est la valeur renvoyée par la commande population(2) ?

$12 \rightarrow p$
$0 \rightarrow j$
$1{,}3 \times 12 - 0 = 15{,}6 \rightarrow p$

$1 \rightarrow j$
$1{,}3 \times 15{,}6 - 1 = \mathbf{19{,}28} \rightarrow p$
renvoyée

```
def population(n) :
    p=12
    for j in range(n) :
        p=1.3*p-j
    return p
```

Que représente le résultat obtenu ?
Calcule et renvoie le 3ème terme de la suite $p_{n+1} = 1{,}3p_n - n$ dont le 1er terme est 12 ($p_0 = 12 \Rightarrow p_2 = 19{,}28$ ou $p_1 = 12 \Rightarrow p_3 = 19{,}28$)

Quelle instruction faut-il saisir pour obtenir un résultat supérieur à 30 ?

3ème boucle : $24{,}064 \rightarrow p$ et 4ème boucle : $30{,}2832 \rightarrow p$ ⇒ **population(4)**

On définit la suite (w_n) par : $\begin{cases} w_0 = 25 \\ w_{n+1} = w_n e^{-n} \end{cases}$

Compléter l'algorithme ci-contre pour que suite(n) renvoie la valeur de w_n pour tout n donné.

Fait bien n calculs, dont le 1er est w_1 donc il arrive à w_n. Pour la 1ère boucle, on a bien pour $k = 0$ et $w_1 = 25 \times e^{-0}$ ça marche…

```
from math import*
def suite(n) :
    w=25
    for k in range(n) :
        w=w*exp(-k)
    return w
```

On définit la suite (u_n) par : $\begin{cases} u_1 = e - 2 \\ u_{n+1} = (n+1)u_n - 1 \end{cases}$

Compléter l'algorithme ci-contre pour que l'instruction suite() renvoie la valeur de u_8 pour tout n donné.

On rentre u_1, le 1er terme calculé va être u_2 et on veut répéter jusqu'à u_8 c'est-à-dire 7 fois le calcul : en partant de 1, il faut s'arrêter à 8.

Pour $u_2 = (1+1)u_1 - 1$ pour $u_3 = (1+2)u_2 - 1$

```
from math import*
def suite() :
    u=exp(1)-2
    for n in range(1, 8) :
        u=(n+1)*u-1
    return u
```

$$\begin{cases} u_0 = 30 \\ u_{n+1} = \frac{1}{2}u_n + 10 \end{cases} \qquad \begin{cases} w_0 = 45 \\ w_{n+1} = \frac{1}{2}w_n + \frac{1}{2}u_n + 7 \end{cases}$$

On a commencé à écrire ci-contre une fonction suite, en langage Python, qui renvoie la valeur du terme w_n pour une valeur de n donnée.

L'exécution de suite(1) ne renvoie pas le terme w_1. Comment modifier la fonction suite afin que l'exécution de suite(n) renvoie la valeur du terme w_n ?

Problème : le w_{n+1} utilise le u_{n+1} pour son calcul, et non le u_n : il faut inverser l'ordre des calculs

```
def suite(n) :
    U=30
    W=45
    for i in range(1,n+1) :
        W=W/2+U/2+7
        U=U/2+10
    return W
```