

Exercice n°1

Cet algorithme, qui est une fonction écrite en langage Python, retourne le plus petit entier naturel n tel que $u_n \geq 45$.

C'est un algorithme de seuil, qui répète le calcul des termes de la suite tant que la condition « être plus petit que 45 strictement » est vérifiée. Dès qu'un terme de la suite ne vérifiera plus $u < 45$ alors la boucle de répétition s'arrêtera et renverra le rang du premier terme qui ne vérifie plus la condition. **REPONSE A**

Exercice n°2

Soit f la fonction définie sur l'intervalle $\left] -\frac{1}{3}; +\infty \right[$ par :

$$f(x) = \frac{4x}{1+3x}$$

On considère la suite (u_n) définie par : $u_0 = \frac{1}{2}$ et, pour tout entier naturel n , $u_{n+1} = f(u_n)$.

1. On a donc pour $n = 0$, $u_1 = f(u_0) = f\left(\frac{1}{2}\right) = \frac{2}{1+\frac{3}{2}} = \frac{2}{\frac{5}{2}} = \frac{4}{5}$.
2. On admet que la fonction f est croissante sur l'intervalle $\left] -\frac{1}{3}; +\infty \right[$.
 - a. On veut montrer par récurrence que, pour tout entier naturel n , on a :

$$\frac{1}{2} \leq u_n \leq u_{n+1} \leq 2.$$

Initialisation : on a $u_0 = \frac{1}{2}$ et $u_1 = \frac{4}{5}$; de plus $\frac{1}{2} \leq \frac{1}{2} \leq \frac{4}{5} \leq 2$, donc :

$\frac{1}{2} \leq u_0 \leq u_1 \leq 2$: l'encadrement est vrai au rang 0 ;

Hérédité : on suppose que pour $n \geq 0$, $\frac{1}{2} \leq u_n \leq u_{n+1} \leq 2$.

La fonction f étant croissante les images des quatre nombres ci-dessus sont rangées dans le même ordre, soit : $f\left(\frac{1}{2}\right) \leq f(u_n) \leq f(u_{n+1}) \leq f(2)$.

Or on a vu que $f\left(\frac{1}{2}\right) = \frac{4}{5}$ et on a $f(2) = \frac{8}{1+6} = \frac{8}{7}$;

de plus $f(u_n) = u_{n+1}$ et $f(u_{n+1}) = u_{n+2}$ donc $\frac{4}{5} \leq u_{n+1} \leq u_{n+2} \leq \frac{8}{7}$.

Or $\frac{1}{2} < \frac{4}{5}$ et $\frac{8}{7} < 2$; on a donc finalement :

$\frac{1}{2} \leq u_{n+1} \leq u_{n+2} \leq 2$: l'encadrement est donc vrai au rang $n+1$.

Conclusion : l'encadrement est vrai au rang 0 et s'il est vrai au rang $n \geq 0$, il est vrai au rang $n+1$: par le principe de récurrence pour tout entier naturel n , on a :

$$\frac{1}{2} \leq u_n \leq u_{n+1} \leq 2.$$

b. La suite (u_n) est croissante et elle majorée par 2, elle est donc convergente vers une limite ℓ telle que : $\frac{1}{2} \leq \ell \leq 2$.

c. La fonction f est continue car dérivable au moins sur $\mathbb{R}_+$ donc la limite ℓ vérifie l'égalité $f(\ell) = \ell$; on résout cette équation :

$$\begin{aligned} f(\ell) = \ell &\iff \frac{4\ell}{1+3\ell} = \ell \iff 4\ell = \ell(1+3\ell) \iff 0 = \ell(1+3\ell-4) \\ &\iff \ell(3\ell-3) = 0 \iff 3\ell(\ell-1) = 0 \iff \begin{cases} \ell = 0 \\ \text{ou} \\ \ell-1 = 0 \end{cases} \iff \begin{cases} \ell = 0 \\ \text{ou} \\ \ell = 1 \end{cases} \end{aligned}$$

Comme $\ell \geq \frac{1}{2}$, la seule solution possible est 1; la suite (u_n) converge vers 1.

3. a. On complète la fonction Python ci-dessous qui, pour tout réel positif E , détermine la plus petite valeur P tel que : $1 - u_P < E$:

```
def seuil(E):
    u = 0,5
    n = 0
    while 1 - u >= E:
        u = 4 * u / (1 + 3 * u)
        n = n + 1
    return n
```

b. On obtient $u_7 \approx 0,999939$, donc $1 - u_7 < 10^{-4}$. Le programme renvoie $n = 7$.

Exercice n°3

1. On a donc $u_1 = 1\,000 \times \left(1 - \frac{10}{100}\right) + 250 = 1\,000 \times 0,9 + 250 = 900 + 250 = 1\,150$.

2. Enlever 10 % c'est multiplier par $1 - \frac{10}{100} = 1 - 0,10 = 0,90$.

Le nombre d'abonnés de l'année précédente est donc multiplié par 0,9; on ajoute ensuite chaque année 250 nouveaux abonnés, donc pour tout naturel n :

$$u_{n+1} = 0,9u_n + 250.$$

3. $u(10)$ donne le nombre d'abonnés au bout de 10 ans; une calculatrice donne $\approx 1\,977$.

Exercice n°4

e. On considère le programme suivant écrit en langage Python :

```
def seuil() :  
    n = 0  
    r = 1  
    while abs(r-sqrt(2)) > 10**(-4) :  
        r = (2+r)/(1+r)  
        n = n+1  
    return n
```

La valeur de n renvoyée par ce programme est 5.

Elle correspond à la plus petite valeur de n pour laquelle la distance entre r_n et $\sqrt{2}$ est inférieure ou égale à 10^{-4} .

Si l'on remplace $10^{**}(-4)$ par 0.005 cela renverra donc la valeur du rang à partir duquel la distance entre r_n et $\sqrt{2}$ sera inférieure à 0,005.

Exercice n°5

Affirmation 1 : FAUX elle contiendra une valeur supérieure strictement à 5000

Affirmation 2 : FAUX, si la limite de la suite est inférieure au nombre par lequel on remplace 5000 alors la boucle ne s'arrêtera jamais car on ne dépassera jamais ce nombre, puisque par définition une limite est une valeur vers laquelle les termes de la suite se rapprochent de plus en plus sans jamais l'atteindre.

Affirmation 3 : FAUX N contiendra la valeur 4.

Exercice n°6

1. Si $u_1 = 0$, on a $u_2 = 2u_1 - 1 = -1$ puis $u_3 = 3u_2 - 1 = -4$ et $u_4 = 4u_3 - 1 = -16 - 1 = -17$ donc $u_4 = -17$.
2. Complétons l'algorithme pour qu'en saisissant préalablement dans U une valeur de u_1 il calcule les termes de la suite (u_n) de u_2 à u_{13} :

```
Pour N allant de 1 à 12  
     $U \leftarrow (N + 1) * U - 1$   
Fin Pour
```

Exercice n°7

$x = x_0$

while $g(x) > 0$:

$x = x + 0.001$

print(x)

Exercice n°8

On considère la suite (U_n) définie par $U_0 = \frac{1}{2}$ et, pour tout entier naturel n , $U_{n+1} = \frac{2U_n}{1+U_n}$.

$$1. U_1 = \frac{2U_0}{1+U_0} = \frac{2 \times \frac{1}{2}}{1 + \frac{1}{2}} = \frac{1}{\frac{3}{2}} = \frac{2}{3}$$

2. On va démontrer par récurrence, pour tout entier naturel n , la propriété $\mathcal{P}_n : U_n = \frac{2^n}{1+2^n}$.

- **Initialisation**

Pour $n = 0$, $\frac{2^n}{1+2^n} = \frac{2^0}{1+2^0} = \frac{1}{1+1} = \frac{1}{2} = U_0$ donc la propriété est vraie au rang 0.

- **Hérédité**

On suppose la propriété vraie pour un entier naturel n quelconque; on va démontrer que la propriété est vraie au rang $n+1$.

$$U_{n+1} = \frac{2U_n}{1+U_n} = \frac{2 \frac{2^n}{1+2^n}}{1 + \frac{2^n}{1+2^n}} = \frac{2 \times 2^n}{1+2^n} \times \frac{1+2^n}{1+2^n+2^n} = \frac{2^{n+1}}{1+2 \times 2^n} = \frac{2^{n+1}}{1+2^{n+1}}$$

Donc la propriété est vraie au rang $n+1$.

- **Conclusion**

La propriété est vraie au rang 0 et elle est héréditaire pour tout $n \geq 0$; donc, d'après le principe de récurrence, la propriété est vraie pour tout entier naturel n .

On a donc démontré que, pour tout entier naturel n , on a $U_n = \frac{2^n}{1+2^n}$.

3. On considère les trois algorithmes suivants dans lesquels les variables n , p et u sont du type nombre. Pour un seul de ces trois algorithmes la variable u ne contient pas le terme U_n en fin d'exécution.

Algorithme 1	Algorithme 2	Algorithme 3
$u \leftarrow \frac{1}{2}$ $i \leftarrow 0$ Tant que $i < n$ $u \leftarrow \frac{2u}{u+1}$ $i \leftarrow i+1$ Fin Tant que	$u \leftarrow \frac{1}{2}$ Pour i allant de 0 à n $u \leftarrow \frac{2u}{u+1}$ Fin Pour	$p \leftarrow 2^n$ $u \leftarrow \frac{p}{p+1}$

Dans l'algorithme 2, le nombre i varie entre 0 et n donc prend $n+1$ valeurs; la valeur de u en sortie est donc U_{n+1} . L'algorithme 2 ne convient donc pas.

Exercice n°9

5. a.

m	a	b	$b - a$	$f(m)$
	2	3	1	
2,5	2	2,5	$0,5 > 0,1$	$\approx 0,26 > 0$
2,25	2,25	2,5	$0,25 > 0,1$	$\approx -1,4 < 0$
2,375	2,375	2,5	$0,125 > 0,1$	$\approx -0,66 < 0$
2,4375	2,4375	2,5	$0,0625 < 0,1$	$\approx -0,22 < 0$