

correction d'évaluation

Suite et récurrence

Soit (u_n) la suite définie par $u_{n+1} = 5u_n + 4$ et $u_0 = 0$.

1. Calculons u_1 , u_2 et u_3 .

$$u_1 = 5u_0 + 4 = 5(0) + 4 = 4$$

$$u_2 = 5u_1 + 4 = 5(4) + 4 = 24$$

$$u_3 = 5u_2 + 4 = 5(24) + 4 = 124$$

2. D'après les premières valeurs de la suite (u_n) , on peut conjecturer qu'elle est croissante.

On considère la propriété $P(n) : \underbrace{u_n}_A = \underbrace{5^n - 1}_B$.

3. Démontrons que la propriété $P(0)$ est vraie, c'est-à-dire que la propriété $P(n)$ est vraie lorsque $n = 0$.

Pour $n = 0$, on a : $A = u_0 = 0$ par hypothèse.

$$B = 5^0 - 1 = 1 - 1 = 0$$

Donc $A = B$. La propriété $P(0)$ est vraie.

4. On suppose que la propriété $P(k)$ est vraie, k étant un nombre entier.

On a donc : $u_k = 5^k - 1$.

5. lorsque la propriété $P(k+1)$ est vraie, on a : $u_{k+1} = 5^{k+1} - 1$.

6. Démontrons que si $P(k)$ vraie, alors $P(k+1)$ vraie.

Supposons $P(k)$ vraie, c'est-à-dire $u_k = 5^k - 1$.

$P(k+1)$ est-elle vraie ? A-t-on $\underbrace{u_{k+1}}_C = \underbrace{5^{k+1} - 1}_D$?

$$\text{On a : } C = u_{k+1} = 5u_k + 4 = 5(5^k - 1) + 4 = 5^{k+1} - 5 + 4 = 5^{k+1} - 1 = D$$

$C = D$, donc la propriété est héréditaire.

7. Les étapes principales mises en jeu dans un raisonnement par récurrence sont l'initialisation et l'hérédité.

8. La propriété $P(n) : "u_n = 5^n - 1"$ est vraie pour tout entier naturel n car l'initialisation est vérifiée dans la question 3) et l'hérédité est vérifiée dans la question 6).

9. On a : $u_5 = 5^5 - 1 = 3125 - 1 = 3124$.

10. On a : $u_{u+1} - u_n = 5^{n+1} - 1 - (5^n - 1) = 5^{n+1} - 1 - 5^n + 1 = 5^{n+1} - 5^n$

$$\text{Donc : } u_{u+1} - u_n = 5^n \times 5 - 5^n \times 1 = 5^n \times (5 - 1) = 4 \times 5^n > 0$$

Donc, pour tout entier n , on a : $u_{u+1} > u_n$ La suite (u_n) est croissante.

Algorithmique et programmation en Python

On considère la suite (u_n) définie par $u_{n+1} = 5u_n + 4$ et $u_0 = 0$ et l'algorithme qui permet d'afficher le terme de rang n de la suite (u_n) .

1. Algorithme corrigé :

```
Entrer n
u ← 0
Pour i allant de 1 à n, faire
    u ← 5u + 4
Afficher u
```

2. Fonctionnement de l'algorithme lorsque l'utilisateur entre la valeur $n = 4$.

n	i	u
4		0
	1	4
	2	24
	3	124
	4	624

3. On considère le code écrit en Python ci-dessous :

```
def u(rang):
    U = 0
    for i in range(rang):
        U = 5*U + 4
    return U

n = int(input("Entrer n : "))
print(u(n))
```

1. Le bloc de code commençant par `def` permet de définir une fonction `u` d'argument `rang` qui renvoie le terme de la suite (u_n) de rang égal à la valeur de l'argument.
2. L'instruction `n = int(input("Entrer n : "))` permet d'inviter un utilisateur à saisir un entier via le message "Entrer n :", ce nombre étant référencé par la variable `n`.
3. Lorsqu'un utilisateur exécute le code et saisit la valeur 3, le terme de rang 3 de la suite (u_n) est affiché à l'écran. Comme $u_3 = 124$, la valeur 124 est affichée.