

Tableur et Python

I - Utilisation du tableau

Exercice 1. On utilise un tableur pour calculer les termes de deux suites u et v .

	A	B	C
1	n	u(n)	v(n)
2		0	100
3		1	110
4		=2*B3-90	190
5		3	170
6		4	290
7		5	
8			

1. D'après la formule saisie en **B4**, et recopiée vers le bas, écrire un procédé pour passer du terme u_1 à u_2
2. Calculer u_4
3. Généraliser ce procédé entre u_{n+1} et u_n
4. On garde le même procédé pour passer de v_{n+1} à v_n . Pourquoi n'obtient-on pas les mêmes résultats ? Argumenter.

Exercice 2. On considère la relation de récurrence :

$$u_{n+1} = \frac{u_n}{2} + 5$$

1. Interpréter par une phrase cette relation entre deux termes consécutifs de la suite u
2. Le terme initial u_1 n'est pas donné mais sera saisi en **B2**. Indiquer la formule à saisir en **B3**, et recopiée vers le , pour obtenir les termes de la suite.
3. Calculer les premiers termes pour :
 - (a) $u_1 = 2$
 - (b) $u_1 = 10$
 - (c) $u_1 = 6$

	A	B
1	n	u(n)
2		1
3		2
4		3
5		4
6		5
7		
8		

Exercice 3. Soit (v_n) la suite définie par $v_0 = 1$ et, pour tout entier naturel n , par $v_{n+1} = 1,02v_n + 100$

1. La suite (v_n) est-elle définie par récurrence ou de manière explicite ?
2. Calculer v_1 et v_2
3. A l'aide de la calculatrice puis du tableur, déterminer une valeur approchée au centième de v_{100}

Exercice 4.

1. Calculer les cinq premiers termes de chaque suite
 - (a) (u_n) définie par $u_n = \frac{1}{n^2} - 7n$, avec $n \in \mathbb{N}^*$
 - (b) (v_n) définie par $v_n = 5(n+2)^2 - 3$, avec $n \in \mathbb{N}$
2. Calculer le dix-huitième terme des deux suites précédentes

Exercice 5.

1. Représenter graphiquement à l'aide du tableur les 5 premiers termes de la suite (u_n) définie pour tout $n \in \mathbb{N}$ par $u_n = n^2 - 2n$
2. Représenter graphiquement à l'aide du tableur les 5 premiers termes de la suite (u_n) définie par $u_0 = 8$ et pour tout $n \in \mathbb{N}$, $u_{n+1} = -\frac{1}{2}u_n + 1$

II - Python : La Boucle For

En python, une boucle permet de répéter plusieurs fois de suite un même traitement.

Lorsque le nombre n d'itérations est connu à l'avance, on parle de **boucle bornée** et on utilise un compteur qui s'incrémente de 1 (généralement) à chaque itération jusqu'à la valeur n.

```
1 for i in range(n):
2     bloc instructions # sera répétée n fois
3     #la variable i prenant successivement
4     les valeurs 0, 1, ..., n-1
```

Exercice 6. Pour calculer un terme de la suite (u_n) définie sur \mathbb{N} , on utilise la fonction Python ci-dessous :

```
1 def u(n):
2     u=12
3     for i in range(n):
4         u=2*u-3
5     return u
```

1. Quelle la valeur de la variable u à la fin du programme pour $n = 3$?
2. Proposer une forme récurrente de la suite (u_n)
3. Réaliser le programme sur le notebook de Basthon et vérifier votre résultat pour $n = 3$

Exercice 7. Pour calculer un terme de la suite (v_n) définie sur \mathbb{N} , on utilise la fonction Python ci-dessous :

```
1 def v(n):
2     v=4
3     for i in range(n):
4         v=2*v-5
5     return v
```

1. Déterminer v_0
2. Proposer une forme récurrente de la suite (v_n)
3. Déterminer la valeur de v_7 en utilisant le notebook de Basthon

Exercice 8. Compléter la fonction somme(n) pour qu'elle renvoie la somme $1 + 2 + 3 + \dots + n$

```
1 def somme(n):
2     S=0
3     for i in range(n+1) : # Pour i allant de 0 à n faire
4         S=.....
5     return(S)
```

Exercice 9. On considère la suite définie pour tout entier naturel n par : $u_n = 2n^2 - n + 1$

1. Ecrire une fonction python rang(n) qui permet de retourner le terme de rang n de cette suite pour une valeur de n donnée
2. Que renvoie l'instruction rang(5) ? Quelle valeur renvoie-t-elle ?