

Table des matières

2	Suites & récurrence	1
	Exercices supplémentaires du chapitre	1
	ES 1 (Influence du premier terme) (Corrigé)	1
	ES 2 (Étude de deux suites récurrentes) (Corrigé)	2
	ES 3 (Recherche d'une expression explicite) (Corrigé)	3
	ES 4 (Suite auxiliaire et somme des termes) (Corrigé)	4
	ES 5 (Suite récurrente d'ordre 2) (Corrigé)	4
	ES 6 (Somme des carrés des premiers entiers) (Corrigé)	5
	ES 7 (Étude d'un sens de variation) (Corrigé)	5
	ES 8 (Expression d'une somme) (Corrigé)	5

Chapitre 2

Suites & récurrence

Exercices supplémentaires du chapitre

 En cliquant sur le lien « **Corrigé** », vous pouvez consulter la correction détaillée de chaque exercice.

ES 1 (Influence du premier terme) (Corrigé)

Partie A

On considère une suite (u_n) définie par $u_0 = a$, où $a \in \mathbb{R}$, et pour tout $n \in \mathbb{N}$, $u_{n+1} = 4u_n - 6$.

1. Recopier et compléter les lignes 4, 7 et 8 de la fonction Python suivante, d'arguments n et a , qui renvoie la liste des $(n + 1)$ premiers termes de la suite (u_n) et en donne une représentation graphique.
2. Conjecturer grâce à ce script le comportement de la suite (u_n) pour les valeurs de a suivantes : $a = 1$ puis $a = 4$.
3. Démontrer les conjectures précédentes.

```
1 from pylab import *
2
3 def liste_termes_u(n,a):
4     u=...
5     L=[u]
6     for i in range(1,n+1):
7         u=.....
8         L.append(...)
9     plot(L,'r.')
10    show()
11    return L
```

Partie B

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

1. Démontrer par récurrence que pour tout $n \in \mathbb{N}$, $1 \leq u_n \leq e^2$.
2. Prouver que la suite (u_n) est croissante.
3. Dans cette question, on s'interroge sur le comportement de la suite (u_n) si l'on choisit d'autres valeurs que 1 pour u_0 . Pour chacune des affirmations suivantes, indiquer si elle est vraie ou fausse, en justifiant.
 - a) Si $u_0 = 10$, alors la suite (u_n) est croissante.
 - b) Si $u_0 = 2$, alors pour tout $n \in \mathbb{N}$, $1 \leq u_n \leq e^2$.
 - c) La suite (u_n) est constante si, et seulement si, $u_0 = 0$.

ES 2 (Étude de deux suites récurrentes) (Corrigé)

Les questions 1 et 2 sont indépendantes.

1. On considère la suite (u_n) définie sur $\mathbb{N}$ par $u_0 = 1$ et pour tout $n \in \mathbb{N}$, $u_{n+1} = u_n + 3^n - 7$.

Pour tout $n \in \mathbb{N}$, on note : $v_n = u_{n+1} - u_n$ et $S_n = v_0 + v_1 + \dots + v_{n-1}$.

- a) Calculer S_n de deux manières différentes.
- b) En déduire que pour tout $n \in \mathbb{N}$, $u_n = \frac{3^n}{2} - 7n + \frac{1}{2}$.
- c) Refaire la démonstration de l'expression de u_n en fonction de n en utilisant un raisonnement par récurrence.
2. On considère la suite (u_n) définie sur $\mathbb{N}$ par $u_0 = 2$ et pour tout $n \in \mathbb{N}$, $u_{n+1} = \frac{1}{5}u_n + 3 \times 0,5^n$.

- a) À l'aide de la calculatrice, compléter le tableau suivant (on arrondira les valeurs à 10^{-2}).

n	0	1	2	3	4	5	6	7	8
u_n	2								

- b) Conjecturer le sens de variation de la suite (u_n) .
- c) Prouver par récurrence que pour tout entier $n \geq 1$, $u_n \geq \frac{15}{4} \times 0,5^n$.
- d) Étudier le signe de $u_{n+1} - u_n$ et en déduire une démonstration de la conjecture de la question **b**).

ES 3 (Recherche d'une expression explicite) (Corrigé)

On considère la suite (u_n) définie par $u_0 = 0$ et pour tout entier naturel n par $u_{n+1} = u_n + 2n + 2$.

Partie A

1. Calculer u_1 et u_2 .
2. On considère les deux algorithmes suivants où la variable n désigne un entier naturel non nul :

Algorithme 1	Algorithme 2
$u \leftarrow 0$	$u \leftarrow 0$
Pour i allant de 1 à n :	Pour i allant de 0 à $n - 1$:
$u \leftarrow u + 2i + 2$	$u \leftarrow u + 2i + 2$
Fin Pour	Fin Pour

De ces deux algorithmes, quel est celui pour lequel la valeur de la variable u à la fin de l'exécution de l'algorithme est égale à u_n ?

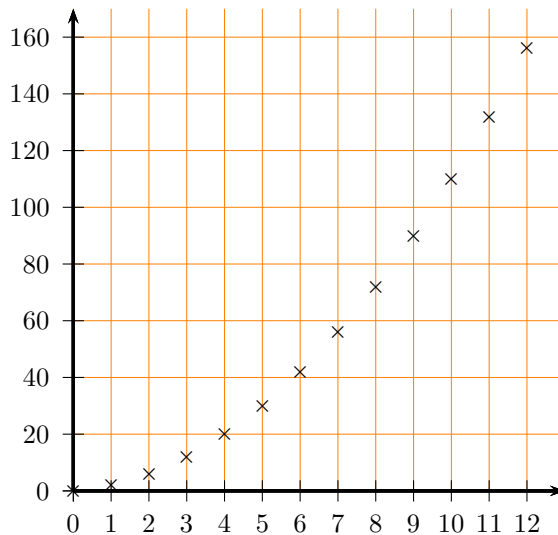
3. Écrire une fonction Python `terme_u`, d'argument `n` un entier naturel, qui renvoie le terme de rang `n` de la suite (u_n) .

Le but des parties B et C suivantes est de proposer deux méthodes permettant d'obtenir une expression explicite de u_n .

Partie B – À partir d'une conjecture graphique

À l'aide de l'algorithme, on a obtenu le tableau et le nuage de points ci-dessous où n figure en abscisse et u_n en ordonnée.

n	u_n
0	0
1	2
2	6
3	12
4	20
5	30
6	42
7	56
8	72
9	90
10	110
11	132
12	156



1. Quelle conjecture peut-on faire quant au sens de variation de la suite (u_n) ?
Démontrer cette conjecture.
2. La forme parabolique du nuage de points amène à conjecturer l'existence de trois réels a , b et c tels que, pour tout entier naturel n , $u_n = an^2 + bn + c$.
Dans le cadre de cette conjecture, trouver les valeurs de a , b et c à l'aide des informations fournies.
3. Démontrer cette conjecture par récurrence et donner une expression explicite de u_n .

Partie C – À l'aide d'une suite auxiliaire

On définit, pour tout entier naturel n , la suite (v_n) par : $v_n = u_{n+1} - u_n$.

1. Exprimer v_n en fonction de l'entier naturel n . Quelle est la nature de la suite (v_n) ?

2. On définit, pour tout entier naturel n , $S_n = \sum_{k=0}^n v_k = v_0 + v_1 + \dots + v_n$.

Démontrer que, pour tout entier naturel n , $S_n = (n+1)(n+2)$.

3. Démontrer que, pour tout entier naturel n , $S_n = u_{n+1} - u_0$, puis exprimer u_n en fonction de n .

ES 4 (Suite auxiliaire et somme des termes) (Corrigé)

Soit (u_n) la suite définie par son premier terme $u_0 = 5$ et, pour tout entier naturel n par $u_{n+1} = 0,5u_n + 0,5n - 1,5$.

1. Dresser le tableau d'état des variables lorsqu'on saisit l'instruction `liste_termes_u(2)` dans la console.

2. Expliquer le rôle de la fonction `liste_termes_u`.

```

1 | def liste_termes_u(p):
2 |     L=[5]
3 |     for i in range(1,p+1):
4 |         L.append(0.5*L[-1]+0.5*(i-1)-1.5)
5 |     return(L)

```

3. La commande `liste_termes_u(4)` saisie dans la console renvoie : `[5, 1.0, -0.5, -0.75, -0.375]`.

Peut-on affirmer, à partir de ces résultats, que la suite (u_n) est décroissante ? Justifier.

4. Démontrer par récurrence que pour tout entier naturel n supérieur ou égal à 3, $u_{n+1} > u_n$.

Que peut-on en déduire quant au sens de variation de la suite (u_n) ?

5. On considère la somme $S_n = \sum_{k=0}^n u_k$. À l'aide de la commande `sum` et de la fonction `liste_termes_u`, proposer une fonction Python `somme_u` d'argument n permettant de calculer S_n puis déterminer S_5 à l'aide de cette fonction.

6. Soit (v_n) la suite définie pour tout entier naturel n par $v_n = 0,1u_n - 0,1n + 0,5$.

Voici un extrait de feuille de tableur :

Quelles formules a-t-on écrites dans les cellules C2 et B3 et copiées vers le bas pour afficher les termes des suites u et v ?

7. Déterminer, en justifiant, une expression de v_n et de u_n en fonction de n uniquement.

8. Démontrer que pour tout $n \in \mathbb{N}$, $S_n = 20(1 - 0,5^{n+1}) + \frac{n^2 - 9n - 10}{2}$.

	A	B	C
1	n	u	v
2	0	5	1
3	1	1	0,5
4	2	-0,5	0,25
5	3	-0,75	0,125
6	4	-0,375	

ES 5 (Suite récurrente d'ordre 2) (Corrigé)

Soit (u_n) la suite définie par $u_0 = 3$, $u_1 = 6$ et, pour tout entier naturel n , $u_{n+2} = \frac{5}{4}u_{n+1} - \frac{1}{4}u_n$.

Partie A

On souhaite calculer les valeurs des premiers termes de la suite (u_n) à l'aide d'un tableur.

On a reproduit ci-dessous une partie d'une feuille de calcul, où figurent les valeurs de u_0 et de u_1 .

1. Donner une formule qui, saisie dans la cellule B4, puis recopiée vers le bas, permet d'obtenir des valeurs de la suite (u_n) dans la colonne B.

2. Compléter le tableau ci-contre. On donnera des valeurs approchées à 10^{-3} près de u_n pour n allant de 2 à 5.

	A	B
1	n	u_n
2	0	3
3	1	6
4	2	
5	3	
6	4	
7	5	

Partie B – Étude de la suite

On considère les suites (v_n) et (w_n) définies pour tout entier naturel n par $v_n = u_{n+1} - \frac{1}{4}u_n$ et $w_n = u_n - 7$.

1. a) Démontrer que (v_n) est une suite constante.

- b) En déduire que, pour tout entier naturel n , $u_{n+1} = \frac{1}{4}u_n + \frac{21}{4}$.

2. a) En utilisant la question 1. b), montrer par récurrence que, pour tout entier naturel n , $u_n < u_{n+1} < 15$.

- b) Que peut-on en déduire pour la suite (u_n) ?

3. a) Démontrer que (w_n) est une suite géométrique dont on précisera le premier terme et la raison.

- b) En déduire que, pour tout entier naturel n , $u_n = 7 - \left(\frac{1}{4}\right)^{n-1}$.

ES 6 (Somme des carrés des premiers entiers) (Corrigé)

On considère la suite (u_n) définie pour tout entier naturel n non nul par $u_n = \frac{6}{n} (1^2 + 2^2 + \dots + n^2) = \frac{6}{n} \sum_{k=1}^n k^2$.

1. Recopier et compléter le script suivant afin que la fonction `liste_carrés` d'argument `n` renvoie la liste des carrés des entiers de 1 à `n`.

1	<code>def liste_carrés(n):</code>
2	<code>liste=[..... for k in range(...,.....)]</code>
3	<code>return liste</code>

À la ligne 2, on dit que la liste `liste` est définie en compréhension.

2. En utilisant la fonction `sum`, écrire une fonction `u`, d'argument `n`, qui renvoie le terme de rang `n` de la suite (u_n) .
3. Créer une liste `L` définie en compréhension qui contient les 30 premiers termes de la suite (u_n) .
4. a) Le script suivant permet d'afficher un nombre donné de points représentant les premiers termes de la suite (u_n) .
Recopier ce script à la suite des précédents.

1	<code>from pylab import *</code>
2	<code>def tracé(nombre_points):</code>
3	<code>for i in range(1,nombre_points+1):</code>
4	<code>plot(i,u(i),'r.')</code>
5	<code>show()</code>

 b) Que doit-on écrire dans la console pour obtenir un graphique représentant les 80 premiers termes de la suite (u_n) ? Visualiser ce graphique.
5. On suppose qu'il existe une fonction f définie sur $[1; +\infty[$ telle que $u_n = f(n)$.
 - a) À l'aide du graphique précédent, quelle conjecture peut-on formuler sur la nature de la fonction f ?
 - b) En utilisant les valeurs de la suite trouvées à l'aide de la liste `L`, conjecturer l'expression algébrique de f . On pourra utiliser le solveur de la calculatrice pour résoudre un système.
 - c) En déduire qu'une formule explicite possible pour (u_n) est : pour tout $n \in \mathbb{N}$, $u_n = (n+1)(2n+1)$.
6. À l'aide d'un raisonnement par récurrence, démontrer que l'on a bien $u_n = (n+1)(2n+1)$ pour tout entier naturel n non nul.
7. En déduire une formule simple donnant la somme des carrés des n premiers entiers naturels non nuls.

ES 7 (Étude d'un sens de variation) (Corrigé)

On considère la suite (u_n) définie sur $\mathbb{N}$ par $\begin{cases} u_0 = 2 \\ u_{n+1} = -\frac{1}{2}u_n^2 + 3u_n - \frac{3}{2} \end{cases}$.

On considère également la suite (v_n) définie pour tout entier naturel n par $v_n = u_n - 3$.

1. Montrer que pour tout entier naturel n , $v_{n+1} = -\frac{1}{2}v_n^2$.
2. Démontrer par récurrence que pour tout $n \in \mathbb{N}$, $-1 \leq v_n \leq 0$.
3. Démontrer que pour tout entier naturel n , $v_{n+1} - v_n = -v_n \left(\frac{1}{2}v_n + 1 \right)$.
4. En déduire le sens de variation de la suite (v_n) puis celui de la suite (u_n) .

ES 8 (Expression d'une somme) (Corrigé)

1. On considère la fonction Python `somme` d'argument n un entier naturel, dont le script est donné ci-contre.

Que renvoient les instructions `somme(3)` et `somme(4)`?

2. Pour tout entier $n \geq 1$, on appelle S_n le résultat renvoyé par l'instruction `somme(n)`.

Montrer que pour tout entier $n \geq 1$, $S_{n+1} = S_n + \frac{n(n+1)}{2}$.

3. Démontrer par récurrence que pour tout entier $n \geq 1$, $S_n = \frac{n^3 - n}{6}$.

1	<code>def somme(n):</code>
2	<code>S=0</code>
3	<code>for i in range(n):</code>
4	<code>S=S+i*(n-i)</code>
5	<code>return S</code>