Entraînement

débranché

papier & crayon

3. Suites et limites

1 On considère la suite (u_n) définie par $u_0 = 0$ et, pour tout entier naturel n, $u_{n+1} = u_n + 3n - 7$.

1. Calculer u_1, u_2, u_3 et u_4.

.....

2. Compléter la fonction rangsuite ci-dessous pour qu'elle affiche le terme de rang n.

```
1  def rangsuite(n):
2      u = 0
3      for k in range (  ):
4          u = ..........
5      return u
```

3. On considère la fonction ci-après écrite dans l'éditeur Python.

```
20 def termesuite(n):
21     return ([rangsuite(k) for k in range (n+1)])
```

Quel sera l'affichage lorsqu'on saisit dans la console Python `>>> termesuite(4)` ?

.....

4. On admet que $\lim\limits_{n\to+\infty} u_n = +\infty$. Écrire une fonction seuil qui permet de déterminer le plus petit entier n tel que $u_n > 1000$.

.....

.....

.....

.....

.....

.....

2 On considère la suite (u_n) définie par $u_0 = 9$, $u_1 = 5$ et, pour tout entier naturel n non nul, $u_{n+1} = \frac{1}{3}(4u_n - u_{n-1})$

1. Calculer u_2, u_3 et u_4. .....

2. Compléter la fonction suiteu ci-contre pour qu'elle affiche le terme de rang n.

```
1  def suiteu(n):
2      u0,u1=9,5
3      for k in range(2,     ):
4          u0,u1=..........
5      return u0
```

3. Que permettent de conjecturer les affichages donnés par les instructions ci-contre écrites dans la console Python ?

.....

```
>>> suite2(10)
3.0001016105268494
>>> suite2(20)
3.000000001720781
```

3 Soit (u_n) la suite géométrique de premier terme $u_0 = 400$ et de raison $q = 0{,}6$.
On pose, pour tout entier naturel n, $S_n = u_0 + u_1 + \ldots + u_n$.

1. Quelle est la limite de la suite (u_n) ?

.....

2. Compléter la fonction sommesuite suivante pour qu'elle affiche le terme de rang n de la suite (S_n).

```
1  def sommesuite(n):
2      u,S=400,400
3      for k in range(1,n+1):
4          u=.....
5          S=.....
6      return ....
```

3. Quelle est la limite de la suite (S_n) ?

.....

4. Écrire une fonction seuilS qui retourne le rang n à partir duquel $S_n > 998$. Cette fonction pourra utiliser sommesuite.

.....

.....

.....

.....

.....

ordinateur

4 **1.** Donner la définition de la suite (u_n) définie par la fonction ci-dessous.

```
def u(n):
    u=0
    for k in range (n):
        u=u+2*k+2
    return u
```

2. Compléter et recopier la fonction plotu qui affiche la représentation graphique des $n+1$ premiers termes la suite (u_n).

```
import matplotlib.pyplot as plt
def plotu(n):
    for k in range(.....):
        plt.plot(....,.....,'r.')
    plt.show()
```

3. En utilisant la fonction plotu, conjecturer le sens de variation de la suite (u_n).

4. Sous quelle forme semble être l'expression explicite de u_n ?

5. On considère la suite (v_n) définie pour tout entier naturel n par $v_n = u_{n+1} - u_n$.
Écrire une fonction v qui prend pour paramètre n et qui affiche les $n+1$ premiers termes de la suite (v_n).

6. Que peut-on conjecturer sur la nature de la suite (v_n) ? Donner son expression explicite.

7. En déduire l'expression explicite de la suite (S_n) définie par $S_n = \sum_{k=0}^{n} v_k$.

8. En déduire l'expression explicite de u_n et démontrer le résultat par récurrence.

5 On se place dans un repère orthonormé et, pour tout entier naturel n, on définit les points (A_n) par leurs coordonnées $(x_n ; y_n)$ de la façon suivante :

$\begin{cases} x_0 = 4 \\ y_0 = 0 \end{cases}$ et, pour tout entier naturel n, $\begin{cases} x_{n+1} = \dfrac{x_n}{2} - \dfrac{y_n}{2} \\ y_{n+1} = \dfrac{x_n}{2} + \dfrac{y_n}{2} \end{cases}$.

1. Compléter la fonction ci-dessous afin de construire les points A_0 à A_n

```
import matplotlib.pyplot as plt
def points(n):
    x,y = 4,0
    plt.plot(x,y,'r.')
    for k in range(.....):
        x,y = ..........
        plt.plot(x,y,'r.')
    plt.show()
```

2. Montrer que pour tout entier naturel n, $A_nA_{n+1} = OA_{n+1}$.

3. Écrire une fonction longueurligne qui prend pour paramètre n et qui retourne la longueur ℓ_n de la ligne brisée $A_0A_1 \ldots A_n$.

4. Conjecturer alors la limite de ℓ_n lorsque n tend vers $+\infty$.

Travaux pratiques

Objectif

- Utiliser des boucles bornées et non bornées

Suites numériques

Suites, limites et recherche de seuils

L'objet du travail ci-dessous est d'étudier l'évolution du stockage des caddies présents en fin de journée dans un supermarché. Ce supermarché dispose sur son parking de trois points d'attache (les points A et B situés aux abords du supermarché et le point C situé au fond du parking).
Pour modéliser les flux de caddies, on fait les hypothèses suivantes.

- Le nombre total de caddies reste constant.
- Chaque jour, le point A perd 12 % de ses caddies mais reçoit 12 % des caddies du point B et 2 % des caddies du point C.
- Chaque jour, le point B perd 12 % de ses caddies mais reçoit 12 % des caddies du point A et 2 % des caddies du point C.
- Chaque jour, le point C perd 4 % de ses caddies.

La veille de l'ouverture du supermarché, il y avait 450 caddies au point A, 250 au point B et 300 au point C.
On désigne par a_n, b_n et c_n les nombres de caddies respectifs présents dans les points A, B et C le soir à la fermeture du supermarché n jours après la veille de l'ouverture.
On admettra, pour l'étude mathématique, que les réels a_n, b_n et c_n peuvent ne pas être entiers.

PARTIE 1 Modélisation

1 Calculer le nombre de caddies présents le soir dans chaque point un jour et deux jours après la veille de l'ouverture.

2 Compléter la fonction caddies qui affiche les représentations graphiques des trois suites ainsi que les nombres de caddies dans chaque point le soir, n jours après la veille de l'ouverture.

```
1  import matplotlib.pyplot as plt
2  def caddies(n):
3      a,b,c = 450,250,300
4      plt.plot(0,a,'r.')
5      plt.plot(0,b,'k.')
6      plt.plot(0,c,'b.')
7      for k in range(1,n+1):
8          a,b,c = ..... , ..... , .....
9          plt.plot(k,a,'r.')
10         plt.plot(.....)
11         .....
12     plt.show()
13     return .....
```

3 Exécuter ce programme plusieurs fois pour différentes valeurs de n.

a. Quelles conjectures peut-on faire sur le sens de variation et les limites des suites ?

b. Quelles sont les valeurs affichées lorsqu'on saisit caddies(100) ?

PARTIE 2 Expressions explicites des suites (a_n), (b_n) et (c_n)

On appelle d_n la différence du nombre de caddies entre les points A et B, n jours après la veille de l'ouverture.

1 Compléter la fonction python suitedn de paramètre n qui retourne les valeurs prises par d_k pour k allant de 0 à n. On pourra faire afficher les termes de la suite arrondis à 10^{-2}.

```
1  def suitedn(n):
2      a,b,c = 450,250,300
3      d = [      ]
4      for k in range (      ):
5          ..........
6          ..........
7      return d
```

2 On a modifié la fonction précédente afin de pouvoir conjecturer la nature de la suite (d_n). Voici l'affichage obtenu dans la console Python lorsqu'on donne à n la valeur 5.

```
>>> suitedn(5)
([200, 152.0, 115.52, 87.8, 66.72, 50.71], [0.76, 0.76, 0.76, 0.76, 0.76])
```

Écrire ci-dessous cette fonction suitedn modifiée.

3 a. Donner les expressions explicites des suites (c_n) et (d_n).

b. En déduire les expressions explicites des suites (a_n) et (b_n) puis leurs limites respectives.

PARTIE 3 Un seuil

1 Écrire une nouvelle fonction nommée seuilcaddies de paramètre M qui retourne le nombre de jours nécessaires après la veille de l'ouverture du supermarché à partir desquels le nombre de caddies dans les points A et B dépassera M.

2 Au bout de combien de jours après la veille de l'ouverture du supermarché y aura-t-il plus de 490 caddies dans les points A et B ?