TP 3 : Boucles *for*

Remarques.

- Ce TP consiste à programmer différentes suites ; on insiste sur ce concept car les algorithmes numériques que l'on verra au deuxième semestre consistent en général à définir une suite et à calculer ses termes ;
- Vouc pourrez utiliser un seul fichier pour écrire toutes les fonctions demandées dans les exercices qui suivent car toutes les suites considérées portent un nom différent ;
- On vous demande explicitement dans chaque exercice de vérifier votre programme en calculant les premières valeurs de la suite, à l'avenir il faudrait penser à faire vous même ce type de vérification. □

Exercice 1 *Suite $u_{n+1} = f(u_n)$.* On considère la suite (u_n) définie par :

$$u_0 = 1$$
$$\forall n \in \mathbb{N},\ u_{n+1} = \frac{u_n(6-u_n^2)}{4}$$

(a) Calculer u_1 (sans l'ordinateur).
(b) Écrire une fonction `u(n)` qui calcule u_n en fonction de n.
(c) Vérifier que `u(1)` donne bien la valeur calculée à la première question et vérifier que, pour n assez grand, `u(n)` est une valeur approchée de $\sqrt{2}$.

Exercice 2 *Suite $v_{n+1} = f(n, v_n)$.* On considère la suite (v_n) définie par :

$$v_0 = 0$$
$$\forall n \in \mathbb{N},\ v_{n+1} = v_n + \frac{(-1)^n}{n+1}$$

(a) Calculer v_1 et v_2 (sans l'ordinateur).
(b) Écrire une fonction `v(n)` qui calcule v_n en fonction de n.
(c) Vérifier que `v(1)` et `v(2)` donnent bien les valeurs calculées à la première question et vérifier que, pour n assez grand, `v(n)` est une valeur approchée de $\ln(2)$.

Exercice 3 *Somme des termes d'une suite.* On considère la suite (w_n) définie par :

$$w_0 = 1$$
$$\forall n \in \mathbb{N},\ w_{n+1} = -\frac{w_n}{(2n+1)(2n+2)}$$

(a) Calculer w_1 et $w_0 + w_1$ (sans l'ordinateur).
(b) Écrire une fonction `w(n)` qui calcule w_n en fonction de n.
(c) Écrire une fonction `s(n)` qui calcule la somme $w_0 + \cdots + w_n$ en fonction de n.
(d) Vérifier que `s(1)` donne bien la valeur $w_0 + w_1$ calculée à la première question et vérifier que, pour n assez grand, `s(n)` est une valeur approchée de $\cos(1)$.

Exercice 4 *Suite $a_{n+2} = f(a_n, a_{n-1})$.* On considère la suite (a_n) définie par :

$$\begin{aligned} a_0 &= 0 \\ a_1 &= 1 \\ \forall n \in \mathbb{N},\ a_{n+2} &= a_{n+1} + a_n \end{aligned}$$

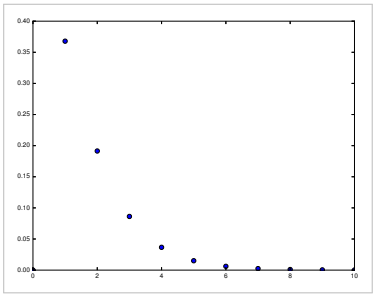
(a) Calculer a_1, a_2, a_3 (sans l'ordinateur).
(b) Écrire une fonction `a(n)` qui calcule a_n en fonction de n.
(c) Vérifier que `a(1)`, `a(2)`, `a(3)` donnent bien les valeurs calculées à la première question et vérifier que, pour n assez grand, a_{n+1}/a_n est une valeur approchée de $(1+\sqrt{5})/2$.

Représentations graphiques

◇ Pour représenter graphiquement une suite, par exemple la suite (u_n) définie par $u_n = \sqrt{n}/e^n$ pour $0 \leqslant n \leqslant 10$, on utilisera les commandes suivantes (dont la signification sera vue ultérieurement) :

```
import matplotlib.pyplot as plt
from math import *
def u(n):
    return n**0.5/exp(n)
plt.plot([u(n) for n in range(11)],'o')
```

```
plt.show()
```



Exercice 5. On considère la suite (u_n) définie par :

$$\begin{aligned} u_0 &= 0 \\ \forall n \in \mathbb{N},\ u_{n+1} &= \frac{3u_n + 2}{u_n + 4} \end{aligned}$$

(a) Écrire une fonction `u(n)` qui calcule u_n en fonction de n.
(b) Représenter graphiquement les premiers termes de la suite (u_n) de l'exercice 1.
(c) Quelle(s) propriété(s) sur la suite (u_n) peut-on conjecturer ?
(d) Les démontrer.

Corrections

Ex 1.

```
def u(n):
    u = 1
    for k in range(n):
        u = u*(6-u**2)/4.0
    return u
print(u(1))
print(u(100))
```

```
1.25 1.414213562373095
```

Ex 2.

```
from math import *
def v(n):
    v = 0
    for k in range(n):
        v = v+float((-1)**k)/(k+1)
    return v
print(v(1),v(2),v(1000),log(2))
```

```
1.0 0.5 0.6926474305598223 0.6931471805599453
```

Ex 3.

```
from math import *
def w(n):
    w = 1
    for k in range(n):
        w = -float(w)/(2*k+1)/(2*k+2)
    return w
def s(n):
    s = 0
    for k in range(0,n+1):
        s = s+w(k)
    return s
print(s(100),cos(1))
```

```
0.5403023058681397 0.5403023058681398
```

Cette manière de faire, pour la fonction **s**, n'est pas très efficace. En effet, pour calculer **s(100)** par exemple, il faut successivement calculer **w(0)**, **w(1)**, etc. jusqu'à **w(100)**. Or, le calcul de **w(k)** demande k étapes de calcul et le calcul de **s(100)** demande alors $1+2+\cdots+100$ étapes de calcul. Plus généralement, le calcul de **s(n)** demande :

$$1+2+\cdots+n=\frac{n(n+1)}{2}$$

étapes de calcul. On peut réécrire la fonction **s** en reprenant à l'intérieur le calcul des w_k :

```
def s(n):
    w = 1
    s = 1
    for k in range(n):
        w = -float(w)/(2*k+1)/(2*k+2)
        s = s+w
    return s
print(s(100),cos(1))
```

```
0.5403023058681397 0.5403023058681398
```

Maintenant, le calcul de **s(n)** demande n étapes de calcul (ce qui est plus efficace que $n(n+1)/2$).

Ex 4.

```
def a(n):
    a = 0
    b = 1
    for k in range(n):
        c = a+b
        a = b
        b = c
    return a
print(a(0),a(1),a(2),a(3))
print(float(a(100))/a(99),(1+5**0.5)/2)
```

```
0 1 1 2 1.618033988749895 1.618033988749895
```

Autre possibilité plus dans l'esprit de PYTHON :

```
def a(n):
    a = 0
    b = 1
    for k in range(n):
        (a,b) = (b,a+b)
    return a
print(a(0),a(1),a(2),a(3))
print(float(a(100))/a(99),(1+5**0.5)/2)
```

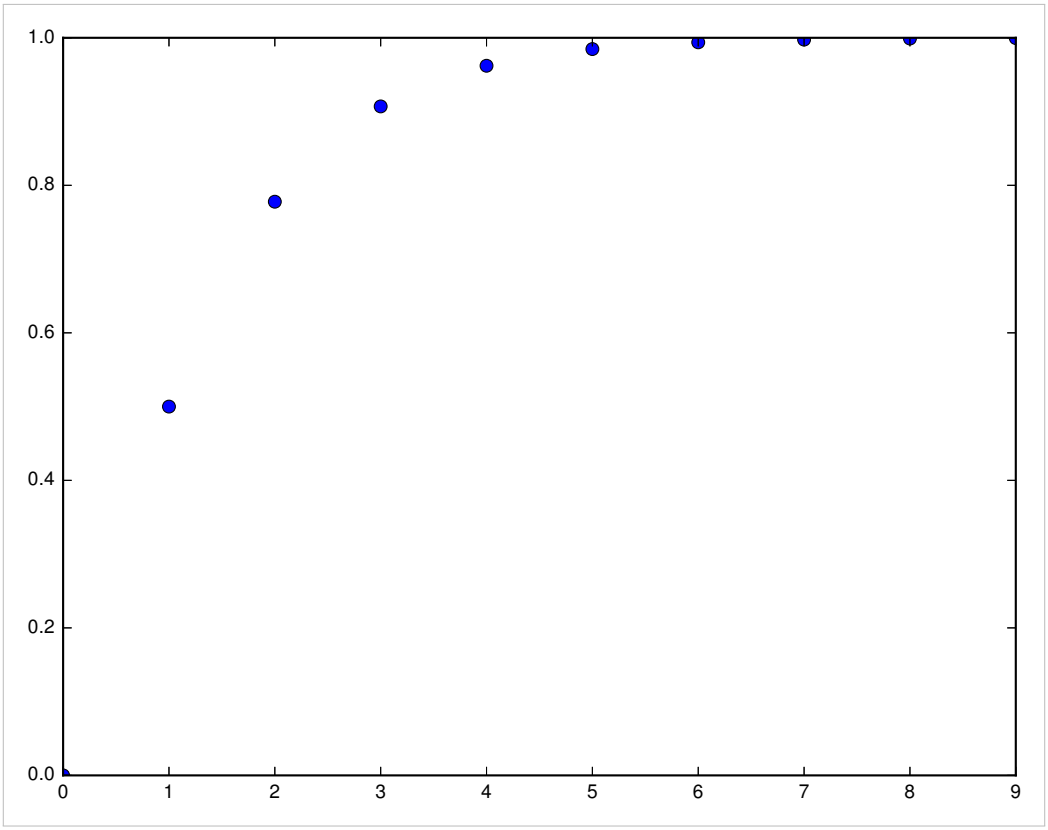
```
0 1 1 2 1.618033988749895 1.618033988749895
```

Ex 5. On définit la fonction **u(n)** :

```
def u(n):
    U = 0
    for k in range(n):
        U = (3*U+2)/(U+4)
    return U
```

On peut maintenant représenter graphiquement les premiers termes de la suite (u_n) :

```
import matplotlib.pyplot as plt
plt.plot([u(n) for n in range(10)],'o')
```



On peut donc faire les conjectures suivantes :

- Les valeurs de la suite (u_n) sont comprises entre 0 et 1 ;
- La suite (u_n) est croissante ;
- La suite (u_n) converge vers 1.

Démontrons ces différents points. On considère tout d'abord pour $n \in \mathbb{N}$ l'hypothèse de récurrence :

$$\mathscr{H}(n) : 0 \leqslant u_n \leqslant 1$$

Comme $u_0 = 0$, $\mathscr{H}(0)$ est vraie. Soit $n \in \mathbb{N}$ et supposons $\mathscr{H}(n)$ vraie, c'est à dire $0 \leqslant u_n \leqslant 1$. Alors $3u_n + 2 \geqslant 0$ et $u_n + 4 > 0$ donc u_{n+1} est bien défini (le dénominateur est non nul) et $u_{n+1} \geqslant 0$ (quotient de deux nombres positifs). Ensuite :

$$u_{n+1} - 1 = \frac{3u_n + 2}{u_n + 4} - 1 = \frac{3u_n + 2 - u_n - 4}{u_n + 4} = \frac{2u_n - 2}{u_n + 4} = \frac{2(u_n - 1)}{u_n + 4}$$

Comme $u_n \leqslant 1$ (hypothèse de récurrence), on a $u_n - 1 \leqslant 0$ et on sait également que $u_n + 4 > 0$ donc $u_{n+1} - 1 \leqslant 0$ c'est à dire $u_{n+1} \leqslant 1$. Par conséquent, $\mathscr{H}(n+1)$ est vraie et, par récurrence, quel que soit $n \in \mathbb{N}$, $0 \leqslant u_n \leqslant 1$. On s'intéresse maintenant à établir la monotonie de la suite (u_n), pour cela on calcule pour $n \in \mathbb{N}$:

$$u_{n+1} - u_n = \frac{3u_n + 2}{u_n + 4} - u_n = \frac{3u_n + 2 - u_n(u_n + 4)}{u_n + 4} = \frac{-u_n^2 - u_n + 2}{u_n + 4} = -\frac{u_n^2 + u_n - 2}{u_n + 4}$$

On considère le polynôme du second degré $P(x) = x^2 + x - 2$, son discriminant est $\Delta = 9$ et on trouve ses racines $x_1 = -2$ et $x_2 = 1$. Ainsi $P(x) = x^2 + x - 2 = (x-1)(x+2)$ et par conséquent :

$$u_{n+1} - u_n = -\frac{(u_n - 1)(u_n + 2)}{u_n + 4}$$

Sachant que $0 \leqslant u_n \leqslant 1$, on a $u_n + 2 \geqslant 0$, $u_n + 4 > 0$ et $u_n - 1 \leqslant 0$. On en déduit que $u_{n+1} - u_n \geqslant 0$ et la suite (u_n) est donc croissante. On sait également que cette suite est majorée (par 1), elle est donc convergente. Notons ℓ sa limite, puisque $0 \leqslant u_n \leqslant 1$ pour tout $n \in \mathbb{N}$, on a $0 \leqslant \ell \leqslant 1$. On sait de plus que :

$$\forall n \in \mathbb{N},\ u_{n+1} = \frac{3u_n + 2}{u_n + 4}$$

et en faisant tendre n vers $+\infty$, on obtient :

$$\ell = \frac{3\ell + 2}{\ell + 4}$$

On en déduit $\ell(\ell + 4) = 3\ell + 2$ ce qui donne après simplification : $\ell^2 + \ell - 2 = 0$. Ainsi, ℓ est une racine du polynôme P définit plus haut, autrement dit $\ell = 1$ ou $\ell = -2$. Sachant que $\ell \in [0, 1]$, on en déduit que $\ell = 1$ et ainsi la suite (u_n) converge vers 1.