T.P. IV - Fonctions & Suites

Code Capytale : `62c9-794137`

I - Fonctions

Solution de l'exercice 1.

1.

```
import numpy as np

def g(x):
    if x < 0:
        return 0
    else:
        return np.exp(-2 * x)
```

2. Utilisez le script précédent pour afficher les valeurs de $g(-1)$ et $g(\ln(2))$.

```
print(g(-1))
print(g(np.log(2)))
```

$\square$

Solution de l'exercice 2.

1.

```
import numpy as np
import matplotlib.pyplot as plt

def f(x):
    return x**n/(1+x)

X = np.linspace(0, 1, 100)

for n in [1, 5, 10, 20, 50]:
    plt.plot(X, f(X), "--", label=r"fn pour n="+str(n))
plt.legend()
plt.show()
```

2. Comme la fonction f_n est positive, l'intégrale I_n est égale à l'aire comprise entre l'axe des abscisses, la courbe représentative de f et les droites d'abscisse $x = 0$ et $x = 1$.

3. En utilisant le graphique, on conjecture que l'aire tend vers 0, soit $\lim_{n \to +\infty} I_n = 0$. $\square$

II - Suites

Solution de l'exercice 3.

1. En factorisant le numérateur, on obtient

$$\begin{aligned} c_n &= 2 - \frac{3^n}{4^{n-1}}\left(1 + \frac{25}{3^n}\right) \\ &= 2 - \left(\frac{3}{4}\right)^n \times 4 \times \left(1 + \frac{25}{3^n}\right). \end{aligned}$$

Comme $3^n \to +\infty$, alors $1 + \frac{25}{3^n} \to 1$.
Comme $\frac{3}{4} \in]0, 1[$, alors $\left(\frac{3}{4}\right)^n \to 0$.
D'après les théorèmes d'addition des limites,

$$\lim_{n \to +\infty} c_n = 2.$$

2. Avant la boucle conditionnelle, `n` contient la valeur 1 et `c` la valeur c_1. Ensuite, on incrémente `n` et on calcule les valeurs de c_n. La boucle s'arrête dès que $c_n \geqslant 1.95$. Ainsi, la valeur renvoyée est le plus petit rang n pour lequel $c_n \geqslant 1.95$. Ce plus petit rang vaut donc 16. $\square$

Solution de l'exercice 4. La variable sert à stocker la valeur de `u` avant qu'on ne la modifie. Ainsi, à l'issue du `i`^{e} passage dans la boucle, `u` contient la valeur de u_i et `v` contient la valeur de u_{i-1}.

```python
v = 0
u = 1
for i in range(2, 11):
    a = u
    u = 4 * u + 2 * v
    v = a

print(u)
```

$\square$

III - Suites et fonctions : la dichotomie

Solution de l'exercice 5.

1. La fonction h est dérivable sur $\mathbb{R}_+^*$ et

$$\begin{aligned} h'(x) &= 3x^2 - \frac{3}{x^4} = \frac{3}{x^4}(x^6 - 1) \\ &= \frac{3}{x^4}(x^3 - 1)(x^3 + 1). \end{aligned}$$

D'une part, $\frac{3}{x^4} > 0$.
D'autre part, comme $x > 0$, alors $x^3 + 1 > 0$.
Enfin, $x^3 - 1 \geqslant 0$ si et seulement si $x^3 \geqslant 1$ si et seulement si $x \geqslant 1$.
On obtient ainsi le tableau de variations suivant :

x	0		1		$+\infty$
$h'(x)$	$\|\|$	$-$	0	$+$	
$h(x)$	$\|\|$	$\searrow$	-1	$\nearrow$	

2. Comme $\lim\limits_{x \to +\infty} x^3 = +\infty$, alors $\lim\limits_{x \to +\infty} \frac{1}{x^3} = 0$. D'après les théorèmes d'addition des limites, $\lim\limits_{x \to +\infty} h(x) = +\infty$.
D'après la définition de h, $h(1) = -1$.

La fonction h est continue et strictement croissante de $[1, +\infty[$ dans $[-1, +\infty[$. Comme $0 \in [-1, +\infty[$, il existe un unique réel α tel que $h(\alpha) = 0$.

3. Comme $h(2) = 2^3 + \frac{1}{2^3} - 3 = 5 + \frac{1}{8}$, alors $\alpha \leqslant 2$. On cherche donc α entre 1 et 2.

```python
def h(x):
    return x**3 + 1/x**3 - 3

a = 1
b = 2
while (b - a) > 10**(-5):
    m = (a + b)/2
    if h(m) * h(a) <= 0:
        b = m
    else:
        a = m

print(a)
```

$\square$

IV - Introduction au produit matriciel

Solution de l'exercice 6.

1. D'après la définition,

$$C_0 = \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

2. D'après la définition des suites (u_n) et (v_n),

$$\begin{aligned} C_{n+1} &= \begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} \\ &= \begin{pmatrix} u_n + v_n \\ 2u_n \end{pmatrix} \\ &= \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix} \begin{pmatrix} u_n \\ v_n \end{pmatrix} \\ &= AC_n. \end{aligned}$$

3. On montre la propriété par récurrence sur n.
Initialisation. Lorsque $n = 0$, montrons que $C_0 = A^0 C_0$.
D'après la définition des puissances de matrices, $A^0 = I$. Ainsi, d'après les propriétés de la matrice identité,

$$A^0 C_0 = I C_0 = C_0.$$

La propriété est donc vraie à l'ordre 0.
Hérédité. Soit $n \in \mathbb{N}$. On suppose que $C_n = A^n C_0$. Montrons que $C_{n+1} = A^{n+1} C_0$. Or,

$$\begin{aligned} C_{n+1} &= A C_n, \text{ d'après la question } \mathbf{2.} \\ &= A \times A^n C_0, \text{ d'après l'hypothèse de récurrence} \\ &= A^{n+1} C_0, \text{ d'après la définition des fonctions puissance} \end{aligned}$$

Ainsi, la propriété est vraie à l'ordre $n+1$.
Conclusion. Finalement, la propriété est vraie à l'ordre 0 et est héréditaire. D'après le principe de récurrence,

$$\forall\, n \in \mathbb{N}, C_n = A^n C_0.$$

4.

```
import numpy as np

n = 12

A = np.array([[1, 1], [2, 0]])

C = np.array([1, 0])

for k in range(1, 13):
    C = np.dot(A, C)

print(C[0], C[1])
```

□