T.P. V - Matrices

Code Capytale : `62c9-794137`

I - Ce qu'il faut savoir

Le module `numpy`, importé via la ligne de commande `import numpy as np` permet de manipuler les matrices avec Python.

* `A = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])` permet de définir une matrice ligne par ligne et d'obtenir ainsi la matrice $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$ qui sera ici stockée dans la variable `A`.
* `3 * A` permet de multiplier `A` par le nombre `3`.
* Si `A` et `B` sont des matrices de mêmes tailles, `A + B` permet d'en calculer la somme.
* Si `A` et `B` sont des matrices de tailles compatibles, `np.dot(A, B)` permet de multiplier les matrices `A` et `B`.

Exercice 1. (Produit matriciel) Soit $A = \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}$. On définit les suites $(u_n)_{n\in\mathbb{N}}$ et $(v_n)_{n\in\mathbb{N}}$ par

$$u_0 = 0,\ v_0 = 1 \text{ et } \forall\ n \in \mathbb{N}, \begin{cases} u_{n+1} & = u_n + v_n \\ v_{n+1} & = 2u_n \end{cases}.$$

Pour tout n entier naturel, on note $C_n = \begin{pmatrix} u_n \\ v_n \end{pmatrix}$.

1. Calculer C_0.

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

3. Montrer par récurrence que, pour tout n entier naturel, $C_n = A^n C_0$.

4. Compléter le script suivant pour qu'il calcule et affiche les termes u_{12} et v_{12}.

```
import numpy as np

n = ...

A = np.array([[...], [...]])

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

for k in range(..., ...):
    C = ...

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