Exercice 1 : Saisie de vecteurs et matrices

a) Saisir les vecteurs suivants grâce à la structure `array` de la librairie `numpy` :

$$u = (5, 8, 7, 9, 30),\ \ v = \underbrace{(2, \cdots, 17)}_{30 \text{ éléments}},\ \ w = \underbrace{(2,\ 2, \cdots, 2,\ 2)}_{100 \text{ éléments}}.$$

b) Saisir les matrices suivantes grâce à la structure `array` de la librairie `numpy` :

$$A = \begin{pmatrix} 4 & 5 & 8 \\ 12 & 5 & 19 \\ 77 & 100 & 10 \end{pmatrix}, B = \begin{pmatrix} 1 & 2 & 3 & \cdots & 50 \\ 5 & 6 & 7 & \cdots & 54 \\ 9 & 10 & 11 & \cdots & 58 \end{pmatrix}, C = \begin{pmatrix} 2 & 0 & \cdots & \cdots & 0 \\ 0 & 4 & 0 & & \vdots \\ \vdots & 0 & 6 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 100 \end{pmatrix}$$

$$\text{et } D = \underbrace{\begin{pmatrix} 1 & 1 & \cdots & 1 \\ 2 & 2 & \cdots & 2 \\ \vdots & & \cdots & \vdots \\ 12 & 12 & 12 & 12 \end{pmatrix}}_{100 \text{ colonnes}}.$$

Exercice 2 : Manipulation des vecteurs

Construire le vecteur X grâce à l'instruction `X=np.random.rand(100)`.

a) Mettre dans la variable n le nombre d'éléments présents dans le vecteur X.

b) Afficher à l'écran la valeur du 3^{eme} élément de X.

c) Construire un vecteur qui contient les éléments $X[8], X[9], \cdots X[20]$.

d) Construire un vecteur qui contient les éléments $X[8], X[10], X[12], \cdots X[20]$.

e) Echanger le 5^{eme} et le 7^{eme} élément (deux façons : une très informatique, l'autre utilisant le slicing).

f) Créer un vecteur Y qui contient tous les éléments de X lus dans le sens inverse.

g) Créer un vecteur Z tel que $Z_i = X_i^2 - 3$.

Exercice 3 : Manipulation de matrices

Les matrices A et B utilisées ici sont les matrices définies à l'exercice 1.

a) Mettre à zéro l'élément $A_{3,3}$ (notation mathématique).
Mettre tous les éléments de la troisième ligne de A à 4.
Créer une matrice $A1$ dont les lignes sont celles de A lues en sens inverse.
Créer une matrice $A2$ en accolant les colonnes 1 et 3 de $A1$ à la droite de A.
Créer une matrice $A3$ constituée des éléments qui se trouvent à l'intersection des deux premières lignes et des deux dernières colonnes de A.

b) Que produisent les instructions $A{*}B$, np.dot(A, B), $A{*}{*}2$, np.dot(A, A), $B{*}{*}2$ et np.dot(B, B) ?

c) Copier la deuxième ligne de A dans un vecteur x. Mettre -5 dans le premier élément de x. Que vaut A à présent ?
Echanger les lignes 1 et 3 de A (deux façons : une très informatique, l'autre utilisant le slicing).
Construire en deux instructions la matrice suivante :

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & a & b & 0 \\ 0 & d & e & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

où a, b, c et d sont des nombres aléatoires.

Exercice 3 : Produit scalaire et norme

Pour chaque question de cet exercice, proposer une solution avec boucle *for* et une solution sans boucle *for*.

a) Effectuer le produit scalaire euclidien entre deux vecteurs x et y donnés.

b) Calculer pour un vecteur v donné les trois normes suivantes :

$$\|v\|_1 = \sum_i |v_i|, \quad \|v\|_2 = (\sum_i |v_i|^2)^{1/2}, \quad \|v\|_\infty = max_i |v_i|.$$

c) Pour un matrice $A = (a_{ij})_{1 \leq i,j \leq n}$ carrée donnée, calculer les normes définies par :

$$\|A\|_1 = max_j \sum_i |a_{ij}|, \quad \|A\|_{fro} = \sqrt{\sum_{i,j} |a_{ij}|^2}, \quad \|A\|_{inf} = max_i \sum_j |a_{ij}|.$$

Exercice 4 : Exercice de vectorisation

Soit X un vecteur de n éléments non nuls $X = (x_1, x_2, \cdots, x_n)$. Pour un n donné, construire le vecteur $P = (p_1, \cdots, p_n)$ tel que

$$p_k = x_1\, x_2 \cdots x_{k-1}\, x_{k+1} \cdots x_n,$$

i.e. le k^{eme} élément de P contient le produit de tous les x_i excepté le k^{eme}.

Exercice 5 : Produit de matrices

Soit A et B deux matrices de taille respective $n \times m$ et $m \times p$.

a) Rappeler l'expression de $C_{i,j}$, l'élément de la ligne i et de la colonne j de C où $C = AB$.

b) Ecrire un programme qui calcule le produit de deux matrices préalablement saisies : on utilisera dans un premier temps des boucles *for* (et aucune fonctionnalité Numpy) puis l'instruction `np.dot`. On pourra travailler avec des matrices aléatoires.

c) Comparer le temps d'exécution de chaque méthode pour de grandes tailles de matrices. On utilisera les instructions :

```
import time
start_time = time.time()
...
interval = time.time() - start_time
```

Faire un graphique qui représente le temps nécessaire pour multiplier deux matrices de taille $n \times n$ en fonction de n.

Exercice 6 : Opérateurs logiques

a) Créer une matrice A de dimension 10×10 et constituée de nombres aléatoires compris entre 0 et 10.

b) Compter le nombre d'éléments de A strictement supérieurs à 5. Multiplier ces nombres par 2.

c) Mettre à 0 les éléments de A compris entre 2 et 8.