

TP 12

Exercice 1

Écrire une fonction **partition(n)** qui construit un vecteur (numpy) x comportant n composantes telles que : $x_0 = 0 \leq x_1 \leq \dots \leq x_{n-1} = 1$ où les x_i , $i \in [[1, n - 2]]$ sont obtenus au hasard sur $[0, 1]$. vous pouvez utiliser les fonctions **random.rand**, **concatenate** et la méthode **sort()** de numpy.

Exercice 2

Écrire une fonction **def math_lapl(n)** : qui construit la matrice d'ordre n suivante :

$$\begin{pmatrix} 2 & -1 & 0 & \dots & 0 \\ -1 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -1 \\ 0 & \dots & 0 & -1 & 2 \end{pmatrix}$$

Exercice 3

Écrire une fonction **def antisym(n)** : qui renvoie une matrice antisymétrique d'ordre n dans laquelle les coefficients situés au dessus de la diagonale sont entrés l'un après l'autre en utilisant la fonction **input** les autres coefficients étant calculés par la fonction.

Exercice 4

On représente une permutation σ de S_n par un tableau A contenant tous les entiers de 1 à n dans un ordre quelconque. $A[i]$ représente $\sigma(i + 1)$. Écrire une fonction **signature(A)** retournant la signature de la permutation représentée par le tableau A .

Exercice 5

Écrire une fonction **def gauss(A,b)** qui résoud un système de Cramer à n inconnues par la méthode du pivot de Gauss. A est la matrice du système et b le second membre est un vecteur colonne, tout comme la solution x , vecteur colonne de n termes. On commencera par écrire un algorithme précisant toutes les étapes du problème.