

## **Arrays Numpy vs listes**

Pour représenter des matrices, il existe plusieurs solutions, l'objectif de ce TP est de comparer l'efficacité des listes (type natif des tableaux redimensionnables en Python) et les Arrays numpy.

- 1) Créer une fonction qui prend trois paramètres  $n$  et  $p$  et  $K$ , entiers et retourne une liste de liste, matrice aléatoire de  $n$  lignes,  $p$  colonnes et contenant des entiers aléatoires compris entre 0 et  $k$ .  
On n'utilisera pas numpy ici mais la bibliothèque random  
Testez votre fonction pour vérifier que les lignes sont bien indépendantes.
- 2) Créez une fonction qui prend les mêmes paramètres et crée un array de dimension  $n,p$  et avec des éléments aléatoires. Avec Numpy tout ceci peut se faire en une seule commande.
- 3) Créez une fonction qui réalise une multiplication entre deux matrices représentées sous forme de liste de liste.
- 4) Avec numpy la multiplication est déjà implémentée.
- 5) Comme dans le TP précédent évaluez les temps de calcul des multiplications pour des matrices aléatoires  $n*n$  avec  $n$  croissant, pour les deux méthodes et tracez sur le même graphique les deux courbes des temps de calcul.
- 6) On considère maintenant une méthode d'ajout qui prend en paramètre une matrice et lui rajoute une ligne en-dessous.  
- implémentez une telle méthode pour les deux représentations des listes et comparez les temps de calcul des deux solutions.  
Commentez le résultat obtenu. Qu'en conclure ?