

## Fiche de TD numéro 4

### Les matrices

Une matrice peut être vue comme une liste de listes (symbolisant chacune une colonne). L'élément de la matrice à l'intersection de la  $i$ ème ligne et de la  $j$ ème colonne sera alors l'élément d'indice  $j$  dans la liste d'indice  $i$ . Cela impose toutefois que toutes les colonnes aient la même taille. On appellera `mat` une instance de la classe `Matrice` dans le reste du sujet.

**Exercice 1 :** Nous allons construire la classe `Matrice`. Spécifier et écrire un constructeur pour cette classe. Il prend en paramètre une liste de listes. Si toutes les lignes n'ont pas le même nombre d'éléments, des zéros seront ajoutés dans la matrice (et uniquement dans la matrice) pour que toutes les lignes aient le même nombre d'éléments.

**Exercice 2 :** Spécifier et écrire les méthodes `mat.nb_lignes` et `mat.nb_colonnes` qui retournent, respectivement, le nombre de lignes et le nombre de colonnes de `mat`.

**Exercice 3 :** Une matrice est dite carrée si elle a le même nombre de lignes que de colonnes. Elle est symétrique si tout élément qui se trouve à l'intersection de la ligne  $i$  et de la colonne  $j$  est égal à celui qui se trouve à l'intersection de la ligne  $j$  et de la colonne  $i$ .

- Spécifier et écrire une fonction `mat.est_matrice_carree()` qui teste si `mat` est carrée.
- Spécifier et écrire une fonction `mat.est_matrice_symetrique()` qui teste si `mat` est symétrique.

**Exercice 4 :** Spécifier puis écrire la fonction `mat.add_mat(mat2)` qui retourne la matrice résultat de l'addition de `mat` et de `mat2`.

**Exercice 5 :** La transposée d'une matrice est la matrice obtenue en inversant ses lignes et ses colonnes. Spécifier et écrire une fonction `mat.transpose_mat()` qui retourne la transposée de `mat`.

**Exercice 6 :** Spécifier puis écrire la fonction `mat.mul_mat(mat2)` qui retourne la matrice résultat de la multiplication de `mat` et de `mat2`.