

## TP 6

# TABLEAUX

### I. Parcours de tableaux

**Exercice 1** Écrire une fonction `moyenne(A)` qui renvoie la moyenne des coefficients du tableau `A`.

**Exercice 2** Écrire une fonction `compartab(A,B)` qui prend en arguments deux tableaux `A` et `B`, et qui renvoie:

- `True` si `A` et `B` ont même taille et mêmes coefficients
- `False` sinon.

### II. Création de tableaux

**Exercice 3** Écrire une fonction `duplique` qui crée une copie d'un tableau donné. Pour cela, on créera un tableau rempli de 0 de même format que le tableau de départ, puis on remplira une par une les cases de ce nouveau tableau avec les éléments du tableau de départ.

**Exercice 4** Écrire une fonction `Pascal(n)` qui récite les `n` premières lignes du triangle de Pascal, où `n` est un entier donné.

**Exercice 5 :**

Écrire une fonction `matid` qui permette de construire la matrice identité de taille  $n \times n$ .

(Que retourne la fonction `eye(n)`?)

**Exercice 6** Écrire une fonction `transposee(A)` qui renvoie la transposée d'une matrice `A`.

**Exercice 7** Écrire une fonction `inverse(A)` qui renvoie l'inverse de la matrice  $A \in \mathcal{M}_2(\mathbb{R})$  si elle est inversible.  
*on pourra utiliser la fonction de l'exercice 10*

### III. Opérations matricielles

**Exercice 8** Pour toute matrice carrée  $A \in \mathcal{M}_n(\mathbb{K})$ , on appelle **trace de `A`**, notée  $\text{tr}(A)$  la somme de coefficients diagonaux de `A`, soit:

$$\text{tr}(A) = \sum_{k=1}^n a_{kk}.$$

Écrire une fonction `mattrace(A)` qui calcule la trace d'une matrice carrée `A`.

(que renvoie la fonction `trace(A)`?)

**Exercice 9** Écrire une fonction `determinant(A)` qui pour toute matrice  $A \in \mathcal{M}_2(\mathbb{R})$  telle que  $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ , calcule et renvoie son déterminant.

**Exercice 10** Écrire une fonction `test_inversible(A)` qui teste si la matrice  $A \in \mathcal{M}_2(\mathbb{R})$  est inversible.  
*on pourra utiliser la fonction de l'exercice 9*

**Exercice 11** Écrire une fonction `produitmat(A,B)` qui calcule le produit matriciel de deux matrices `A` et `B` et renvoie un message d'erreur quand le produit n'est pas possible.