



6 TP : Les Tableaux

6.1 Gérer un tableau

1. Écrire une fonction `Afficher_Tab(T, N)` qui prend en entrée un tableau `T` et sa taille `N` et qui affiche les `N` éléments du tableau `T`.

Exemple :

Le tableau `T = {43, 55, 67, 79}` sera affiché par l'appel `Afficher_Tab(T, 4)`; comme suit :

```
43    55    67    79
```

2. Écrire une fonction `Saisir_Tab` qui prend en entrée deux paramètres ; un tableau `T` d'entiers ainsi que sa taille `N` et qui demandera à l'utilisateur de saisir `N` entiers qu'elle stockera dans le tableau `T`.

Exemple :

`Saisir_Tab(T, 4)`; pourrait s'exécuter comme suit :

Merci de saisir les 4 elements de votre tableau :

```
T[0] : 43
```

```
T[1] : 55
```

```
T[2] : 67
```

```
T[3] : 79
```

3. Écrire la fonction `Moyenne_Tab` qui calcule la moyenne des `N` éléments d'un tableau `T` d'entiers. `N` et `T` sont fournis comme paramètres; la moyenne est retournée comme résultat.
4. A l'aide des fonctions ci-dessus, écrire un programme qui saisit un tableau `A` d'une taille inférieure ou égale à 100 et affiche le tableau ainsi que la moyenne de ses éléments.

6.2 Autour du Maximum

1. Écrire une fonction `Maximum` qui indique, à la fonction appelante, la plus grande valeur, son indice dans le tableau ainsi que son adresse dans la mémoire.
2. Écrire la fonction `main` pour tester la fonction `Maximum` .

6.3 Saisie et affichage des matrices

1. Écrire la fonction `Saisir_Dimensions` à quatre paramètres `L`, `L_Max`, `C`, `C_Max` qui lit les dimensions `L` et `C` d'une matrice à deux dimensions. Les dimensions `L` et `C` doivent être inférieures à `L_Max` respectivement `C_Max`.
2. Écrire la fonction `Saisir_Matrice` à trois paramètres `M`, `L`, `C`, qui lit les éléments d'une matrice `M` d'entiers et de dimensions `L` et `C`.

3. Écrire la fonction `Afficher_Matrice` à trois paramètres `M`, `L`, `C` et `C_Max` qui affiche les éléments de la matrice `M` de dimensions `L` et `C`.
4. Écrire un programme pour tester les fonctions ci-dessus.
5. Réécrire les deux fonctions `Saisir_Matrice` et `Afficher_Matrice` ci-dessus en notation pointeur (bien choisir les paramètres nécessaires).

6.4 Quelques opération sur les matrices

1. Écrire une fonction `Somme_Matrice` qui effectue l'addition de deux matrices `A` et `B`.
2. Écrire une fonction `Tanspo_Matrice` qui effectue la transposition d'une matrice `M`.
3. Écrire un programme qui permet de tester les fonctions ci-dessus.

6.5 Carrés magiques

Un carré magique est une matrice carrée de taille $n \times n$ telle que la somme de chaque rangée, de chaque colonne et de chaque diagonale principale soient égales. Un carré magique est dit normal s'il contient chaque entier compris entre 1 et n^2 exactement une fois.

Exemples. Le tableau suivant est un carré magique normal :

6	7	2
1	5	9
8	3	4

1. Écrire une fonction `Carre` qui retourne 1 si le tableau passé en paramètre avec ses dimensions est une matrice carrée (qui a autant de lignes que de colonnes), 0 sinon.
2. Écrire deux fonctions `Somme_Ligne` et `Somme_Colonne` qui prennent en entrée un tableau et un numéro de ligne (respectivement de colonne) et qui renvoient la somme des éléments de cette dernière.
3. Écrire deux fonctions `Somme_Diagonale` et `Somme_AntiDiagonale` qui retournent la somme de la diagonale (respectivement de l'antidiagonale) du tableau passé en paramètre.
4. Écrire une fonction `Carre_Magique` qui retourne 1 si le tableau passé en paramètre est un carré magique normal et 0 sinon.
5. Écrire un programme qui demande à l'utilisateur de saisir un tableau, et affiche s'il s'agit d'un carré magique normal.