



## 6 TD : Les Tableaux

☞ **Remarque.** Pour les exercices suivants, on ne demande ni la saisie des tableaux ni leur affichage.

### 6.1 Sommes de deux tableaux

Écrire une fonction qui prend en entrée trois tableaux d'entiers de même taille et qui range dans le troisième la somme des éléments des deux tableaux passés en paramètres.

**Exemple.**

T1 : 

2	0	5	9	3	6	8	1
---	---	---	---	---	---	---	---

T2 : 

5	2	8	1	9	3	4	7
---	---	---	---	---	---	---	---

T3 : 

7	2	13	10	12	9	12	8
---	---	----	----	----	---	----	---

### 6.2 Insertion d'un élément dans un tableau

1. Écrire une fonction `Est_Trié` qui retourne 1 si les  $N$  éléments du tableau sont en ordre croissant (au sens large) et 0 sinon.
2. Écrire une fonction `Position` qui retourne la position d'insertion éventuelle d'une valeur  $v$  dans un tableau supposé trié. La fonction doit retourner l'indice de la première valeur  $v$  dans le tableau si elle est présente, ou l'indice où devrait se trouver  $v$  dans le tableau si elle n'existe pas.
3. Écrire une fonction `Décaler` qui prend en entrée un tableau  $T$  d'entiers, sa taille  $L$  et une position  $i$  de type entier ( $L$  et  $i$  doivent être strictement inférieurs à 100). Cette fonction décale les éléments du tableau  $T$  d'une position vers la droite à partir de l'indice  $i$ .

**Exemple :** avec  $L = 8$  après `Decaler(T,L,3)`,  $L$  devient égale à 9.

T : 

3	8	11	17	48	56	61	90
---	---	----	----	----	----	----	----

 → T : 

3	8	11	17	17	48	56	61	90
---	---	----	----	----	----	----	----	----

4. Écrire une fonction `Inserer_Element` qui prend en entrée un tableau  $T$  d'entier à une dimension (**supposé déjà trié dans l'ordre croissant ↗**), sa taille  $L$  (qui doit être strictement inférieur à 100) et un entier  $N$ . Cette fonction permet d'insérer l'élément  $N$  dans le tableau  $T$  à la bonne position pour que le tableau reste trié.

**Exemple :** avec  $L = 8$  après `Inserer_Element(T,L,24)`,  $L$  devient égale à 9.

T : 

3	8	11	17	48	56	61	90
---	---	----	----	----	----	----	----

 → T : 

3	8	11	17	24	48	56	61	90
---	---	----	----	----	----	----	----	----

5. Écrire une fonction `main` pour tester si un tableau est trié puis d'insérer un nouvel élément saisi par l'utilisateur dans ce tableau s'il est trié.

### 6.3 Multiplication de deux matrices

Écrire une fonction permettant de multiplier une matrice A de dimensions N et M avec une matrice B de dimensions M et P et qui retourne la matrice C de dimensions N et P :

$$A(N,M) * B(M,P) = C(N,P)$$

La multiplication de deux matrices se fait en multipliant les composantes des deux matrices lignes par colonnes :

$$C_{ij} = \sum_{k=1}^{k=m} (a_{ik} * a_{kj})$$

**Rappel :**

$$\begin{array}{|c|c|c|} \hline a & b & c \\ \hline e & f & g \\ \hline h & i & j \\ \hline k & l & m \\ \hline \end{array} * \begin{array}{|c|c|} \hline p & q \\ \hline r & s \\ \hline t & u \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline a*p + b*r + c*t & a*q + b*s + c*u \\ \hline e*p + f*r + g*t & e*q + f*s + g*u \\ \hline h*p + i*r + j*t & h*q + i*s + j*u \\ \hline k*p + l*r + m*t & k*q + l*s + m*u \\ \hline \end{array}$$

### 6.4 Recherche de Points-cols

Écrire un programme permettant de rechercher dans une matrice donnée A les éléments qui sont à la fois un maximum sur leur ligne et un minimum sur leur colonne. Ces éléments sont appelés des points-cols. Afficher les positions et les valeurs de tous les points-cols trouvés.

**Exemples :** A[1][1] = 3 est un point-cols :

3	2	1
6	5	4
9	8	1

**Méthode :** Établir deux matrices d'aide MAX et MIN de même dimensions que A, telles que :

$$MAX_{ij} = \begin{cases} 1 & \text{si } A_{ij} \text{ est un maximum sur la ligne } i \\ 0 & \text{sinon} \end{cases}$$

$$MIN_{ij} = \begin{cases} 1 & \text{si } A_{ij} \text{ est un minimum pour la colonne } j \\ 0 & \text{sinon} \end{cases}$$