

## Série de TD N 02 : listes chaînées

**Exercice 01 :** Soit L une liste chaînée d'entiers.

Ecrire l'algorithme permettant de saisir une liste chaînée de N entiers et ensuite affiche ses éléments, sa longueur, la somme de ses éléments, son minimum, et son miroir. L'algorithme doit libérer tout l'espace mémoire réservé par la liste L.

### Exercice 02

Écrire les opérations récursives suivantes sur les listes chaînées d'entiers.

1. **Appartient (L, x)** permettant de chercher un élément dans la liste L.
2. **Max (L)** qui retourne le plus grand élément de L.
3. **Est\_triée (L)** permettant de vérifier si une liste est triée dans un ordre croissant ou non.

**Exercice 03:** écrire une fonction (récursive et itérative) Fusion (L1, L2) qui prend deux listes triées dans l'ordre croissant et retourne une liste triée, dans le même ordre, contenant les deux listes.

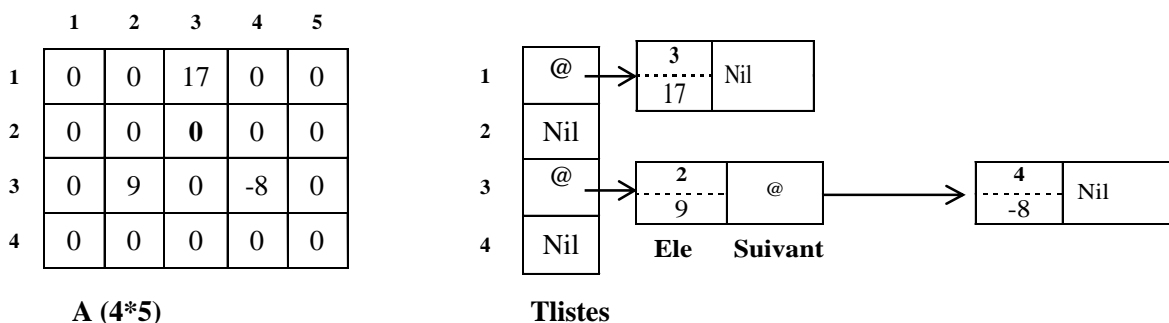
**Exercice 04:** écrire une version itérative et une autre récursive pour les fonctions suivantes sur les listes chaînées d'entiers.

1. Insérer (pos, x, L) : permettant d'insérer x dans la position pos.
2. Supprimer\_pos (L, pos), permettant de supprimer l'élément existant dans la position pos.
3. InsérerTrie (L, x) permettant d'insérer x dans la liste triée L tout en gardant l'ordre des éléments.

### Exercice 05

Une matrice creuse est une matrice dont la majorité des éléments sont nuls. On peut représenter une matrice creuse, en ne tenant compte que des éléments non nuls, par l'utilisation d'un tableau de N listes (N est le nombre de lignes de la matrice). Chaque ligne de la matrice est représentée par une liste linéaire chaînée ordonnée (selon le rang de la colonne) des éléments non nuls. Chaque élément de la liste contient l'indice (j) de la colonne et la valeur (v) de l'élément non nul de la matrice.

**Exemple :** sous-dessous un exemple sur la représentation d'une matrice A par un tableau de listes Tlistes.



1. Ecrire une procédure ***afficher*** (**Tlistes**, **N**, **M**) admettant en entrée un tableau *Tlistes* représentant une matrice creuse de  $N \times M$  éléments et qui affiche tous les éléments, nuls et non nuls, de cette matrice.
2. Ecrire une fonction ***pourcentage*** (**Tlistes**, **N**, **M**) admettant en entrée un tableau *Tlistes* représentant une matrice creuse de  $N \times M$  éléments et qui retourne le pourcentage des éléments nuls dans la matrice.
3. Écrire une fonction ***diagonale*** (**Tlistes**, **N**) admettant en entrée un tableau **Tlistes** représentant une matrice carrée de  $N \times N$  éléments et qui permet de vérifier si cette matrice est diagonale ou non. Une matrice carrée est dite diagonale si tous les éléments hors la diagonale sont nuls.

### **Exercice 05: supplémentaire**

Soit L une liste chaînée d'étudiants pour le module ASDD3. Où chaque étudiant est représenté par son numéro, son nom, son prénom, une note TD, une note TP et une note contrôle.

1. Faire le découpage modulaire nécessaire et écrire l'algorithme permettant de saisir une liste de N étudiants et ensuite affiche la liste des étudiants qui ont obtenu le module.
2. Ecrire la fonction Ajourné (L) permettant de créer une liste contenant les étudiants qui n'ont pas obtenu le module.

### **Exercice 7 : supplémentaire**

Nous considérons la construction d'une liste des nombres premiers inférieurs ou égaux à un entier n donnée. Pour construire cette liste, on commence, dans une première phase, par y ajouter tous les entiers de 2 à n en commençant par le plus grand et en terminant par le plus petit qui se trouvera a la tête de la liste. On considère ensuite successivement les éléments de la liste dans l'ordre croissant en on supprime tous leurs multiples stricts. Écrire le programme et les fonctions nécessaires.