

TD N°4

Objectifs : savoir manipuler et utiliser les structures des données listes linéaires (SDLL)

Exercice 1 : Inversion d'une liste chaînée

Écrire une procédure qui permet d'inverser une liste chaînée Ls dans une liste Ls_Inv.

Exercice 2: Tri par bulles d'une liste chaînée

Écrire une procédure qui permet de trier une liste chaînée Ls dans l'ordre croissant des éléments (utiliser la méthode de tri par bulles).

Exercice 3 : Parcours d'une liste chaînée

| | | |
|-----------------|-----------------|-------------------|
| Etudiant=struct | Cellule =Struct | Liste=struct |
| Nom :chaine | Cle :Etudiant | Premier :^Cellule |
| Note :réel | Suiv :^Cellule | Dernier :^Cellule |
| finStruct | finStruct | finStruct |

On suppose qu'on a en entrée une liste d'étudiants, dont un champ s'appelle note. Calculer la moyenne de la classe (on suppose que la liste est non vide).

Exercice 4 : Insertion dans une liste chaînée

On suppose qu'on a en entrée une liste triée par ordre croissant d'entiers. Ecrivez l'action qui ajoute un entier dans la bonne position de façon que la liste reste encore triée après l'insertion.

Exercice 5 : Tri par fusion d'une liste chaînée

Proposer un algorithme qui fait la concaténation de deux listes (en entrée) retournant une liste (en sortie). Par exemple, si l'on a les listes (3, 7, 2) et (5, 4) on obtient la liste (3, 7, 2, 5, 4).

Proposer ensuite un algorithme séparant une liste en deux. Etant donné une liste en entrée, un élément sur deux va dans la première liste et un élément sur deux va dans la deuxième liste.

Faire une action de fusion de listes supposées triées par ordre croissant en une troisième liste triée.

Faire, en utilisant au choix les trois actions précédentes une action de tri fusion qui étant donné deux listes en entrée (pas forcément triées) retourne une liste triée.