

Listes Linéaires Chainées (Travail personnel)

Exercice 1 Interclassement de n listes linéaires chaînées

Soit T un tableau de n listes linéaires chaînées ordonnées.

1. Transformer T en une Liste L de listes sans création de nouvelles listes
2. Donner la procédure qui affiche tous les éléments de L .
3. Créer une liste ordonnée LD contenant tous les éléments de T en parcourant parallèlement les n listes de T (Interclassement)
4. Eliminer les doublons dans LD .

Exercice 2 Matrices creuses

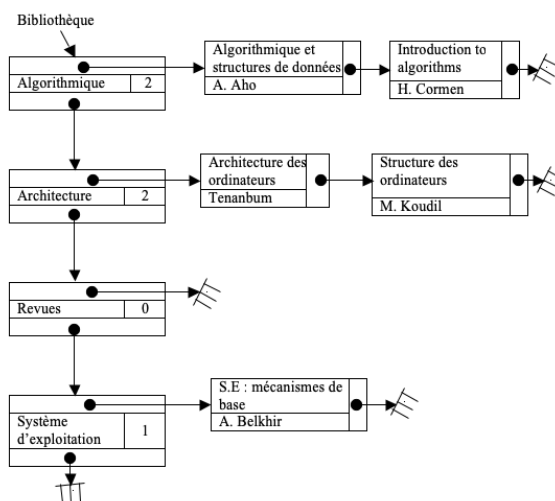
Une matrice est dite creuse lorsque le nombre d'éléments nuls y figurant est très supérieur à celui des éléments non nuls. On peut représenter une matrice creuse en ne tenant compte que des éléments non nuls. Chaque ligne de la matrice est une liste linéaire chaînée ordonnée (selon le rang de la colonne) des éléments non nuls. Une table de N éléments (N étant le nombre de lignes de la matrice) donne les adresses de tête de chacune des listes. Un élément de la liste contient l'indice de la colonne et la valeur de l'élément.

1. Donner la déclaration de la structure de données nécessaire.
2. Ecrire la procédure permettant de Remplir une telle structure à partir d'une matrice $A(M, N)$ donnée.
3. Ecrire la procédure permettant de calculer la somme de deux matrices ainsi représentées.

Sachant qu'un élément de la matrice occupe 4 octets, qu'un pointeur occupe 2 octets et qu'un indice occupe 2 octets,

1. quel est le gain d'espace quand le nombre de zéros est de μ
2. quelle valeur minimale attribuée à μ pour que l'on puisse opter pour une telle représentation

Exercice 3 Dans cet exercice, un étudiant en informatique souhaite représenter sa bibliothèque personnelle en utilisant une structure dynamique. La structure proposée est représentée dans la figure suivante :



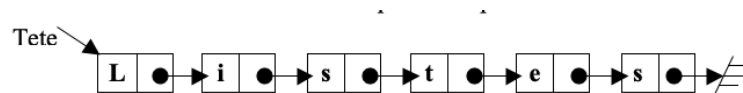
La liste verticale contient les catégories des livres avec le nombre de livres dans chacune, tandis que les listes horizontales contiennent les titres des livres avec leurs auteurs dans chaque catégorie.

1. Donner la déclaration des structures de données nécessaires à l'implémentation de cette bibliothèque ainsi que la procédure d'initialisation de ces structures.
2. Ecrire la procédure d'ajout d'une nouvelle catégorie (l'ajout se fait toujours à la fin de la liste).
3. Ecrire la procédure d'insertion d'un nouveau livre (l'ajout se fait au début de la liste)
4. Ecrire la procédure qui permette d'afficher les livres d'une catégorie donnée.
5. Ecrire la fonction qui retourne le nombre total de livre dans la bibliothèque.
6. Ecrire la procédure qui permette de supprimer une catégorie avec tous ses livres.

Exercice 4 On souhaite représenter les chaînes de caractères par des listes linéaires chaînées où chaque caractère d'une chaîne est rangé dans un maillon de la liste :

Exemple :

La chaîne de caractères "Listes" est représentée par la liste suivante :



Il est demandé d'écrire en langage algorithmique vu dans le cours les fonctions suivantes :

1. **Fonction NbreOccur(L :Pointeur(TMaiillon), C : caractère) : entier**
Qui retourne le nombre d'occurrences du caractère C dans la chaîne de caractères représentée par la liste de tête L.
2. **Fonction Concat(L1, L2 : Pointeur(TMaiillon)) : Pointeur(TMaiillon) ;**
Qui retourne la liste représentant le résultat de concaténation des deux chaînes représentées par L1 et L2.
3. **Fonction Retirer(L :Pointeur(TMaiillon), C :caractère) :Pointeur(TMaiillon) ;**
Qui retourne une liste égale à la liste L sans le caractère C.
4. **Fonction Position(L :Pointeur(TMaiillon), C :caractère) :Pointeur(TMaiillon) ;**
Qui retourne un pointeur sur la première occurrence du caractère C dans la liste L.
5. **Fonction Remplacer(L :Pointeur(TMaiillon), C1,C2 :caractère) : Pointeur(TMaiillon) ;**
Qui retourne une liste égale à L avec toutes les occurrences de C1 remplacées par C2.

Exercice 5 Pour éviter les débordement de calcul des opérations arithmétiques sur les grands nombres naturels, on convient de les représenter par des listes linéaires chaînées où chaque chiffre d'un nombre est rangé dans un maillon de la liste.

Exemple :

Le nombre "309876309876" est représenté par la liste suivante :



Écrire les fonctions suivantes :

1. **Fonction Pair (L :Pointeur(TMaiillon)) : Booleen**
Qui retourne Vrai si le nombre représenté par L est pair, et Faux sinon.
2. **Fonction Somme(L1,L2 : Pointeur(TMaiillon)) : Pointeur(TMaiillon) ;**
Qui retourne une nouvelle liste représentant la somme des deux nombres représentés par L1 et L2.
3. **Fonction Mult(L1, L2 :Pointeur(TMaiillon)) : Pointeur(TMaiillon) ;**
Qui retourne une nouvelle liste représentant le résultat de la multiplication des deux nombres représentés par L1 et L2.
4. **Fonction Div(L1, L2 :Pointeur(TMaiillon)) : Pointeur(TMaiillon) ;**
Qui retourne une nouvelle liste représentant le résultat de la division du nombre représenté par L1 sur le nombre représenté par L2.
5. **Fonction Mod(L1, L2 : Pointeur(TMaiillon)) :Pointeur(TMaiillon) ;**
Qui retourne une nouvelle liste représentant le reste de la division du nombre représenté par L1 sur le nombre représenté par L2.