

Programmation Impérative 2 - L2 Informatique

TP3 - Piles/Files

Exercice I -

Implémenter les TAD `Pile` et `File` à l'aide de listes chaînées. (pensez à la libération de la mémoire) Vérifiez que vos programmes marchent bien à l'aide d'un programme de test utilisant successivement plusieurs insertions, suppressions, affichages.

Pour agrémenter cet exercice, vous pouvez prévoir un module d'affichage (basé sur la bibliothèque MLV) qui affiche la liste en dessinant une case pour chaque cellule des flèches pour les relations définies par les pointeurs, vous pouvez indiquer les adresses mémoires réelles utilisées.

Exercice II - On désire implémenter la gestion d'une messagerie électronique à l'aide d'une file. De manière simplifiée, pour chaque message on stockera le nom de l'expéditeur, le sujet, la date de réception et le contenu du message; chacun de ces éléments pourront être représentés sous la forme d'une chaîne de caractères au sein d'une structure adéquate.

En vous servant du TAD `File`, implémenter l'ajout de messages, la suppression, l'affichage de message selon la date de réception, l'expéditeur et le sujet (par ordre alphabétique). Vous implémenterez un TAD spécifique pour la gestion des messages.

Implémenter le même procédé à l'aide d'une représentation à base d'un tableau.

Exercice III - Calcul d'expressions avec notation postfixée (ou polonaise). La notation polonaise permet de définir des expressions mathématiques sans utiliser de parenthèse. Par exemple,

$3 * (4 + 7)$ peut s'écrire en notation postfixée sous la forme $3\ 4\ 7\ +\ *$.

Autre exemple le calcul $((1 + 2) * 4) + 3$ pourra se représenter soit sous la forme $1\ 2\ +\ 4\ *\ 3\ +$ soit sous la forme $3\ 4\ 1\ 2\ +\ *\ +$.

Ecrire un programme capable d'évaluer une expression fournie en notation postfixée à l'aide d'une pile. On supposera que l'expression est entrée sur la ligne de commande et ne contient que des entiers.

Exercice IV - (Bonus) Implémenter la représentation par matrice creuse vue en cours. Programmer l'addition de matrices. Vérifiez que ces fonctions marchent bien en comparant les résultats avec la représentation usuelle en deux dimensions.

Implémenter une fonction de libération de mémoire pour le type matrice creuse.