

T.P. 7

Listes chaînées et pointeurs

1 Listes chaînées

1.

Ecrire une procédure récursive recherchant le plus grand et le plus petit élément d'un tableau de n éléments.

2.

Ecrire une version itérative de la procédure précédente.

3.

Ecrire une fonction insérant un nouvel élément dans une liste chaînée avec un élément de référence.

4.

Etant données deux listes chaînées l1 et l2, écrire une fonction qui détermine si l2 est une sous-liste de l1.

5.

Ecrire une procédure qui supprime toutes les occurrences d'un élément x d'une liste chaînée.

6.

Ecrire une procédure qui libère la mémoire occupée par une liste chaînée.

7.

Ecrire les fonctions de gestion d'une pile chaînée, puis d'une pile contigüe :

- pile_vide
- creer_pile
- depiler
- dernier
- empiler

8.

Trouver comment ranger deux piles dans un seul tableau de sorte qu'on ne puisse plus empiler sur aucune que si le tableau est plein. Ecrire une fonction `empiler(x, tab, b)` qui empile l'élément `x` sur l'une des deux piles dans `tab` selon la valeur du booléen `b`.

9.

Le miroir d'une chaîne de caractères `s` est la chaîne obtenue par lecture de `s` de droite à gauche. Par exemple, `miroir(fois)=soif`. Programmer la fonction `miroir` de façon récursive et itérative.