



## Travaux Pratiques N° 4

### Les Listes Chainées

#### EXERCICE 1 :

On souhaite classer par ordre alphabétique une liste de personnes dont on a stocké le nom et le prénom. On respecte les règles de classement suivantes :

- les personnes sont classées suivant leurs noms de famille,
- si deux personnes ont le même nom de famille, elles sont classées d'après leur prénom.

La structure de données suivante permet de stocker les renseignements concernant une personne:

```
typedef struct personne {  
    char *nom;  
    char *prenom;  
} personne;
```

1. Écrire une fonction `change_chaine()` qui met en majuscule le premier caractère d'une chaîne de caractères passée en paramètre et en minuscules le reste des caractères de cette chaîne. Écrire une fonction `change_personne()` qui applique la conversion précédente aux champs d'une variable `personne`.
2. Écrire une fonction `int compare_personnes(personne p1, personne p2)` qui compare deux variables personnes comme indiqué dans l'énoncé. La valeur de retour de cette fonction aura la même signification que celle de la fonction `strcmp()`:
  - `< 0` si `p1` est situé avant `p2`,
  - `> 0` si `p2` est situé avant `p1`,
  - `== 0` si `p1` et `p2` sont identiques.

On supposera que les champs `nom` et `prenom` ont d'abord été convertis comme à la question précédente.

3. Écrire une fonction `echange_personnes()` qui échange les contenus de deux variables de type `personne`.
4. Écrire une fonction `tri_personnes()` qui trie un tableau de données de type `personne` comme indiqué dans l'énoncé (on utilisera un tri facile à implémenter. . .).
5. Écrire une fonction qui récupère une chaîne de caractère saisie au clavier, le caractère `'\n'` indiquant la fin de la saisie (on pourra bien sûr réutiliser la fonction de l'exercice précédent. . .).
6. Écrire une fonction `main()` qui effectue la saisie d'un tableau de personnes (il faudra demander la taille du tableau ou bien indiquer un moyen d'arrêter la saisie des noms et prénoms), convertit les prénoms et les noms comme à la première question, trie le tableau et l'affiche.



### EXERCICE 2:

Ecrire un programme C qui permet de **créer** et de **lire** une liste chaînée d'entiers puis **afficher** ses éléments.

La structure utilisée sera de la forme:

- un entier
- un pointeur sur la structure de la liste.

### EXERCICE 3 :

On souhaite maintenant gérer une bibliothèque permettant de contenir un nombre de livres maximum non prédéfini. On veut de plus concevoir une structure permettant de faciliter le classement des livres dans la bibliothèque.

#### 1. Structure

- a. Expliquez pourquoi une liste chaînée est une solution élégante à ce problème.
- b. Ecrivez la structure maillon d'une telle liste, chaque maillon représentant un livre de la bibliothèque. On supposera qu'un chaînage simple est suffisant.

#### 2. Ajout d'un Livre

Proposez une fonction permettant de saisir puis d'ajouter un livre à n'importe quel endroit de la bibliothèque

#### 3. Fonction de recherche

Ecrivez une fonction qui affiche tous les livres dont l'année d'édition est supérieure ou égale à 2000.

#### 4. Fonction de suppression d'un livre

Proposez une fonction qui permettrait de supprimer un livre de la bibliothèque. On ne tolérera pas les fuites de mémoire.

### EXERCICE 4:

1. En C, les chaînes de caractères ne sont pas des listes chaînées, mais nous pouvons arranger ça : écrivez les définitions nécessaires pour représenter une chaîne de caractères comme une liste chaînée où chaque maillon contient un unique caractère.
2. Écrivez ensuite une fonction qui transforme une chaîne tableau en une chaîne liste, et une fonction qui fait l'inverse. Testez-les.

### EXERCICE 5:

On considère la structure Employe définie par :

- Matricule : un entier
- Nom : chaîne de caractères
- Prénom : chaîne de caractères
- Ancienneté : un entier

Ecrire un programme qui permet de :

1. Saisir une liste chaînée de 5 employés

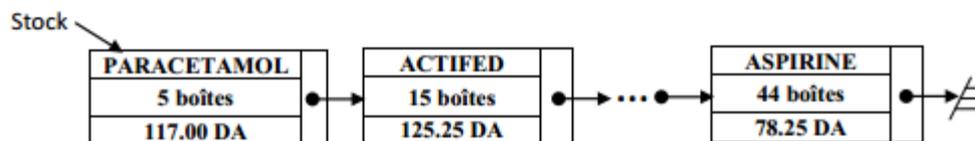


2. Ajouter un employé à la fin de la liste
3. Supprimer l'employé le plus ancien
4. Afficher la liste ainsi obtenue

#### EXERCICE 6 :

Un pharmacien souhaite traiter les informations concernant son stock de médicaments par ordinateur. On vous propose de représenter ces informations sous forme de liste chaînée où chaque maillon contient le libellé d'un médicament, la quantité disponible (nombre de boîtes) et le prix unitaire.

#### Exemple :



On vous demande de :

1. Donner les structures de données nécessaires à la représentation de ce stock.
2. Ecrire la procédure **Vendre(Med, NbBoites)** permettant de retirer, si possible, 'NbBoites' du médicament 'Med' du stock.  
**Attention :** Il faut supprimer du stock le médicament dont la quantité atteint 0.
3. Ecrire la procédure **Acheter(Med, NbBoites, Prix)** permettant au pharmacien d'alimenter son stock par 'NbBoites' du médicament 'Med' ayant le prix unitaire 'Prix' DA. On considère qu'un médicament prenne toujours le nouveau prix.
4. Ecrire la fonction **PrixStock** permettant de calculer le prix total des médicaments dans le stock.

#### EXERCICE 7:

Le problème de **Joséphus Flavius** : Dans un bureau de recrutement,  $n$  personnes numérotées de 1 à  $n$  sont disposées en cercle suivant l'ordre de leur numéro. Le chef de bureau est au centre, puis se dirige vers la personne n°1. Sa stratégie est d'éliminer chaque deuxième personne qu'il rencontre en tournant sur le cercle ; la dernière personne restante est embauchée. Par exemple s'il y a 10 personnes,  $n=10$ , les personnes 2, 4, 6, 8, 10, 3, 7, 1, 9 sont éliminées et la personne restante est le n°5.



Faire un programme de simulation :

1. Pour n entré par l'utilisateur donner le numéro de la personne restante.
2. Au lieu de prendre chaque deuxième personne généraliser en prenant la k-ème personne, k entré par l'utilisateur.

Il s'agit de faire une liste circulaire. Chaque élément est une personne (nom, numéro)

#### EXERCICE 8 :

On représente un polynôme par une liste chaînée. Chaque cellule de la liste correspond à un monôme, avec son coefficient et son exposant. Par exemple, sur la figure ci-dessous, on représente la liste chaînée correspondant au polynôme  $10x^7 + 5x^4 + 3x^2 + 5$ .

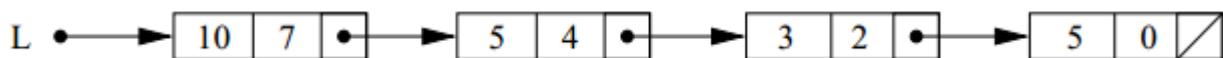


FIGURE 1: La liste correspondant au polynôme  $10x^7 + 5x^4 + 3x^2 + 5$

**Exercice 1** Écrire une fonction qui prend en paramètre un polynôme et rajoute un monôme à la liste. Le coefficient et l'exposant du monôme seront passés en paramètre. On supposera que le monôme est de degré supérieur au degré du polynôme passé en paramètre.

**Exercice 2** Écrire une fonction de saisie au clavier d'un polynôme.

**Exercice 3** Écrire une fonction d'affichage d'un polynôme.

**Exercice 4** Écrire une fonction de destruction d'un polynôme.

**Exercice 5** Écrire une fonction qui recopie un polynôme.

**Exercice 6** Écrire une fonction non préservative de dérivation d'un polynôme.

**Exercice 7** Écrire une fonction d'addition de deux polynômes.

**Exercice 8** Écrire une fonction de multiplication d'un polynôme par un monôme.

**Exercice 9** Écrire une fonction de multiplication de deux polynômes.