

Examen de Fin de Semestre

Exercice 1 :

Parmi tous les entiers supérieurs à 1 et inférieurs à 500, il y en a seulement 4 qui peuvent être représentés par la somme des cubes de leurs chiffres ($xyz = x^3 + y^3 + z^3$).
Ecrire un programme qui détermine ces 4 nombres.

Exercice 2 :

- 1) Ecrire une fonction de nom `AffListe` qui affiche les éléments de la liste chaînée passée en paramètre. Vous donnerez une version itérative et une version récursive de cette fonction.
- 2) Ecrire une fonction de nom `AffListeInv` qui affiche, dans l'ordre inverse du chaînage, les éléments d'une liste chaînée passée en paramètre (La liste possède un seul point d'entrée qui est la tête). Pour la liste chaînée : 4 1 7 3 `AffListeInv` devra produire la sortie : 3 7 1 4
- 3) Ecrire une fonction qui inverse l'ordre du chaînage des éléments dans une liste. (Cette technique est implémentée par certains interpréteurs pour l'opération de Ramasse Miettes).

Exercice 3 :

L'Algérie dispose de 3 opérateurs de téléphonie mobile : *Mobilis*, *Djezzy* et *Nedjma*. Chaque opérateur détient une liste de ses clients. Un client est défini par un nom, numéro de la carte nationale (entier) et le numéro de téléphone (de la ligne).

- 1) Ecrire une fonction qui permet d'ajouter un client à la liste des clients d'un opérateur. L'ajout se fait selon la politique des files d'attente (FIFO).
- 2) Ecrire une fonction qui permet de rechercher un abonné chez un opérateur (la recherche se fait selon le numéro de la carte nationale qui est unique).
- 3) Construire une liste des clients qui sont abonnés chez les trois opérateurs en même temps. (On suppose que les trois listes sont pointées par les têtes *Mob*, *Dje* et *Ned*).

Chaque opérateur décide de mettre ses clients fidèles (qui ne sont pas chez les autres opérateurs) dans une liste à part.

- 4) Ecrire une fonction qui permet d'éclater une liste d'abonnés d'un opérateur en 2 listes : celle des clients fidèles et celle des clients non fidèles.

L'ARPT est un organisme chargé de gérer les 3 opérateurs. Pour ce faire, il détient une pile où chaque élément est composé de : nom de l'opérateur, un indicatif, le nombre d'abonnés et un champ qui contient un pointeur vers la liste des abonnés.

- 5) Donner la déclaration des structures de données pour une telle implémentation.
- 6) Donner la fonction qui permet d'ajouter (empiler) un opérateur à la pile.
- 7) Trier la pile par nombre d'abonnés décroissants.



Examen

Exercice 1 : (12 points)

Soit T un tableau à N lignes et M colonnes. En supposant que le tableau est initialisé, nous voulons à partir de l'élément $T(i,j)$ (i et j sont inconnus), afficher tous les éléments de T , chacun doit être affiché une seule fois.

Pour ce faire, à partir de l'élément de départ $T(i,j)$, qui est d'abord affiché, on procède par l'affichage de ses quatre voisins $T(i-1,j)$, $T(i,j-1)$, $T(i+1,j)$, $T(i,j+1)$ à condition qu'il ne soient pas déjà affichés. Chacun des quatre voisins sera considéré comme un élément de départ et la procédure sera répétée. Et ainsi de suite jusqu'à ce que tous les éléments soient affichés.

A titre d'exemple (voir tableau ci-dessous), on affichera $T(i,j)$, $X1$, $X2$, $X3$, $X4$, $Y1$, $Y2$, $Y3$,

...

Proposez la structure de données adéquate à cette stratégie et écrire le programme en C permettant de résoudre ce problème.

			Y1		
		Y2	X1	Y3	
		X2	T(i, j)	X4	
			X3		

Figure 1. Exemple du parcours des éléments du tableau T

Exercice 2 : (08 points)

Soit CH une chaîne de caractères représentant un nombre écrit en octal (exemple : $CH = "027"$, la chaîne doit commencer par zéro). Ecrire une fonction récursive en langage C qui permet de calculer à partir de CH le nombre décimal correspondant. Dans l'exemple cité, CH sera convertie au nombre 23.

En déduire la version itérative de cette fonction.

Prof. SLIMANE LARABI

<http://perso.usthb.dz/~slarabi/accueil.html>