

LO21- TP n°2

À l'assaut des pointeurs & Listes chaînées

Exercice n°1 :

Écrire une fonction de permutation circulaire de trois variables.

Exercice n°2 :

Écrire une fonction qui prend comme arguments a , b , c de type **float**, et des pointeurs vers un entier n , et deux flottants $x1$ et $x2$. Après l'appel de la fonction, $*n$ doit contenir le nombre de solutions de l'équation $ax^2 + bx + c = 0$ dans \mathbb{R} , et $*x1$, $*x2$ cette ou ces solution(s).

Exercice n°3 :

Manipulation d'une liste chaînée de Produit

Le but de cet exercice est d'écrire en langage C des fonctions de traitements d'une liste chaînée de produit. Les prototypes de ces fonctions seront déclarés dans un fichier header nommé *liste.h* et les fonctions elles-mêmes dans un fichier *liste.c*.

Écrire les fonctions pour réaliser les manipulations suivantes :

- Ajouter un produit en tête de la liste.
- Ajouter un produit en fin de liste.
- Supprimer un produit en tête de la liste.
- Supprimer un produit en fin de la liste.
- Rechercher un produit dans une liste avec son codeP.
- Afficher tous les produits.

Fichier Header : *liste.h*

```
#ifndef __LISTE_H__
#define __LISTE_H__

#include <stdlib.h>
#include <stdio.h>

// Declaration de la liste pour produit
typedef struct prod {
    int codeP;
    int quantP ;
    float prixP;
    struct prod *next;
} prod;

typedef prod *List;

List ajout_tete(List l, int c , int q , float p);
List ajout_fin(List l, int c , int q , float p);
List supp_tete(List l);
List supp_fin(List l);
void recherche(List l, int c) ;
void afficher_list(List l);

#endif
```

Fichier : *liste.c*

```
#include "liste.h"

//Commentaire
List ajout_tete(List l, int c , int q , float p){

    .....
}

//Commentaire
List ajout_fin(List l, int c , int q , float p) {

    .....
}

//Commentaire
List supp_tete(List l) {

    .....
}

//Commentaire
List supp_fin(List l) {

    .....
}

//Commentaire
void recherche(List l, int c) {

    .....
}

//Commentaire
void afficher_list(List l) {

    .....
}
```

Exercice n°4 :

Compilation avec la commande make

Ecrire un programme *tp2main.c* pour tester toutes les fonctions déclarées dans l'exercice n°1.

Le fichier *tp2main.c* fait appel au header *liste.h* pour utiliser les fonctions de traitement de la liste produit. Pour compiler le programme *tp2main.c*, vous devez utiliser la commande *make* qui va compiler tous les fichiers (.c et .h) nécessaires. La compilation avec *make* se fait avec un fichier sans extension nommé *Makefile* qui contient la syntaxe suivante :

Fichier : *Makefile*

```
tp2main: liste.o tp2main.o
    gcc -o tp2main liste.o tp2main.o

liste.o: liste.c
    gcc -o liste.o -c liste.c -W -Wall -ansi -pedantic

tp2main.o: tp2main.c liste.h
    gcc -o tp2main.o -c tp2main.c -W -Wall -ansi -pedantic
```

Exercice n°5 :

Ecrire un code qui permet de déclarer une liste contenant les produits ci-dessous :

Produit				
code	3	5	7	6
quantite	0	1	2	10
prix	5.8	12.4	13.5	10

Compléter le programme de manière qu'il affiche les produits dont la quantité est inférieure à un seuil donné.