

Série N° 1

Exercice 1 : (Gestion d'une pharmacie)

On souhaite gérer une **pharmacie** ayant une liste de **clients** et une liste de **médicaments** en créant des structures **C (Client, Medicament et Pharmacie)** :

- le client est représenté par une structure **Client** ayant un nom (**chaîne**) et un crédit (**double**). Le crédit représente la somme que **doit** le client à la pharmacie. Le crédit peut être négatif si le client a versé plus d'argent que le montant à payer.
- le médicament est représenté par une structure **Medicament** ayant un libelle (**chaîne**), un prix (**double**) et une quantité en stock (**int**).
- la pharmacie est représenté par une structure **Pharmacie** ayant un tableau de client (**tabC**) de taille **MAXC** et un tableau de médicament (**tabM**) de taille **MAXM**. La pharmacie est définie aussi par le nombre de clients ajoutés à la pharmacie (**nbC**) et le nombre de médicaments ajoutés à la pharmacie (**nbM**).

- Donner la définition des structures **Client**, **Medicament** et **Pharmacie**:
- En utilisant les structures définies, écrire un programme **main** qui permet de :

- Créer 2 clients et les initialisés avec les données suivantes :

	nom	credit
Client c1 ;	Hamid	300
Client c2 ;	Aicha	-400

- Créer 3 médicaments et les initialisés avec les données suivantes :

	libelle	prix	qteEnStoc
Medicament m1 ;	Catafla	49	14
Medicament m2 ;	Dolipran	22	16
Medicament m3 ;	Baycute	25	6

- Créer 1 pharmacie et l'initialisé avec les données suivantes :

	Solde
Pharmacie ph ;	3000

- Ajouter les clients à la pharmacie.
- Réaliser un achat des médicaments par la pharmacie.
- Afficher l'ensemble des clients à partir de la pharmacie créée.
- Afficher l'ensemble des médicaments à partir de la pharmacie créée.
- Réaliser un achat de la pharmacie du médicament m1.
- Réaliser un vent de la pharmacie du médicament m2 au client c1.
- Afficher l'ensemble des clients à partir de la pharmacie créée.
- Afficher l'ensemble des médicaments à partir de la pharmacie créée

Exercice 2 :

Soit la structure Matrice représentant un type matrice d'entier:

```
typedef struct Matrice {  
    int nbLignes ;  
    int nbColonnes;  
    int ** mat;  
} Matrice;
```

Ecrire un programme qui :

1. Crée une matrice A et lui alloue l'espace de mémoire nécessaire.
2. Initialise la matrice A par des données saisies par l'utilisateur.
3. Affiche la matrice A.
4. Libère l'espace mémoire de la matrice A.

Exercice 3 : (Gestion de la pharmacie)

Reprendre l'exercice 1 en transformant le tableau automatique de client et le tableau de médicament de la structure Pharmacie sous forme d'un pointeur qui sera alloué dynamiquement au besoin avec une taille définie par l'utilisateur.

La structure Pharmacie aura donc la forme suivante :

```
typedef struct Pharmacie {  
    double solde;  
    Client * tabC;  
    int MAXC;  
    int nbC;  
    Medicament * tabM;  
    int MAXM;  
    int nbM;  
} Pharmacie;
```