

## Série TP02 : Les enregistrements

### Exercice N°1

On considère le programme c suivant :

```
#include <stdio.h>
typedef struct {
char nom[20];
short qunt;
float prix;
} produit;
int main( ){
return 0; }
```

1. Dans le programme principal, déclarer une variable nommée p de type produit.
2. Initialiser les champs membres de variable p par des valeurs introduites au clavier.
3. Afficher à l'écran la valeur de chaque champ membre de variable p.
4. Dans le programme principal, déclarer un tableau TAB des produits de taille cinq éléments.
5. Initialiser les champs membres de chaque élément du tableau TAB par des valeurs introduites au clavier.
6. Afficher à l'écran les valeurs de chaque élément du tableau TAB.

### Exercice N°2

Définir un enregistrement nommé "Livre" qui contient trois champs membres : un titre dont la taille maximale est 30 caractères, un nom d'auteur avec une taille maximale de 20 caractères, et une année de publication.

1. Écrire une procédure nommée "saisirlivre" qui demande à l'utilisateur de saisir au clavier les informations d'un seul livre passé comme paramètre.
2. Écrire une procédure "infolivre" qui affiche à l'écran les informations concernant un seul livre passé comme paramètre.

3. Écrire une procédure nommée "remplivres" qui utilisera la procédure "saisirlivre" pour remplir toutes les informations concernant un tableau des livres passé comme paramètre avec leur taille.
4. Écrire une procédure nommée "afflivres" qui utilisera la procédure "infolivre" pour afficher à l'écran tous les livres en détails provenant d'un tableau passé comme paramètre avec leur taille.
5. Écrire le programme principal "main" qui fait appel aux sous programmes définis précédemment.

### Exercice N°3

Définir un enregistrement nommé "étudiant" composé par les champs nom, âge, filière et un tableau de notes.

1. Écrire une fonction nommée "makeetudiant" qui fait allouer un espace mémoire de type etudiant et retourne son adresse mémoire.
2. Écrire une procédure "initetudiant" qui admet comme argument un pointeur sur etudiant pour lire toutes les informations relatives sur un etudiant au clavier.
3. Écrire une procédure nommée "viewetudiant" qui admet un pointeur sur etudiant pour afficher à l'écran ses informations.
4. Écrire une fonction nommée "moyenne" qui admet un pointeur sur etudiant pour calculer et renvoyer sa moyenne des notes. Sachant que toutes notes sont de mêmes coefficients.
5. Écrire une procédure "tabetudiant" qui admet un pointeur sur tableau d'étudiants, permettant d'ajouter un étudiant d'une façon ordonnée (ordre croissant) selon l'alphabet de champ nom. Puis, d'afficher à l'écran le nouveau contenu du tableau.
6. Écrire le programme principal "main" qui fait appel aux sous programmes définis précédemment.