

Séance 5: Structures

Une structure est un type qui permet de stocker plusieurs données dans une même variable. Ces données peuvent être du même type ou de types différents.

Exemple :

```
struct point
{
float x,y ;
};
```

La déclaration d'une variable de type struct point se fait comme suit : **struct point p** ; Les données d'une structure sont accessibles à l'aide de l'opérateur « . ». Par exemple pour accéder aux coordonnées d'un point p on met p.x et p.y

Pour éviter la répétition du mot struct lors de déclaration des variables on peut définir un raccourci par un **typedef** lors de la définition de la structure. Par exemple si on définit un type Point3D :

```
typedef struct point
{
float x,y,z ;
}Point3D ;
```

La déclaration des variables de type Point3D se fait : **Point3D p** ;

Exercice 1 :

Ecrivez les fonctions suivantes :

- void affiche(Point3D p) qui permet d'afficher un Point3D sous la forme (x,y,z)
- Point3D saisiePoint3D() qui permet de saisir les coordonnées au clavier
- Point3D addition(Point3D p1, Point3D p2) qui additionne les coordonnées de deux Point3D
- Une fonction main pour tester les fonctions précédentes

Exercice 2 :

Définir une structure NombreRationnel permettant de coder un nombre rationnel avec numérateur et dénominateur. On écrira des fonctions de saisie, d'affichage, de multiplication et d'addition de deux rationnels.