



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**
Membre de **HONORIS UNITED UNIVERSITIES**

1ère **AP**

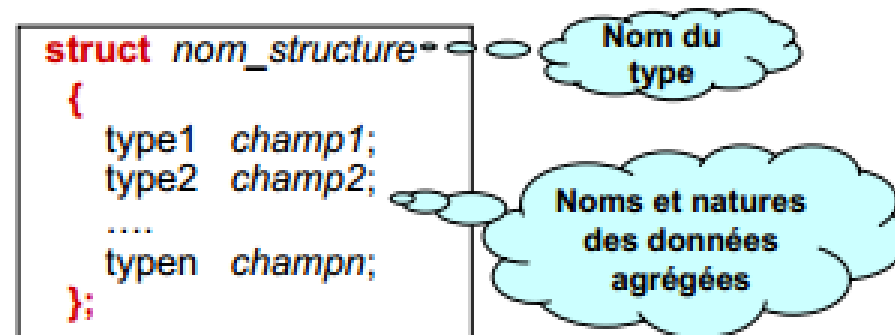
Langage de Programmation 2:
Les Structures

Pr. JORIO Ali

a.jorio@emsi.ma

Structures

- La **structure** permet de désigner sous un nom un ensemble de valeurs pouvant être de types différents.
- L'accès à chaque élément de la structure (nommé **champ**) se fera, cette fois, non plus par une indication de position (indice), mais par son nom au sein de la structure.



Remarque : Le mot **struct** fait partie du nom du type

Déclaration d'une structure

- Exemple de declaration :

```
Struct enreg{  
    int numero ;  
    int qte ;  
    float prix ;  
};
```

- Cette déclaration définit un **modèle de structure** mais ne réserve pas de variables correspondant à cette structure. Ce modèle s'appelle ici **enreg** et il précise le nom et le type de chacun des champs constituant la structure.
- Une fois un tel modèle défini, on peut déclarer des variables du type correspondant

```
    struct enreg art1 ;
```

Réserve un emplacement nommé art1 << de type enreg >> destiné à contenir deux entiers et un flottant

Utilisation d'une structure

- En C, on peut utiliser une structure de deux manières :
 - en travaillant individuellement sur chacun de ses champs
 - en travaillant de manière globale sur l'ensemble de la structure
- Chaque champ d'une structure peut être manipulé comme n'importe quelle variable du type correspondant.



- `art1.numero = 15;`
- Il est possible d'affecter à une structure le contenu d'une structure définie à partir du même modèle :
 - `art1 = art2;`

Définir des synonymes avec typedef

- Le mot clé **struct** fait partie intégrante du nom du type de structure. On peut nommer un type de structure en utilisant le mot clé *typedef*.
- En faisant usage de *typedef*, les déclarations des structures **art1** et **art2** peuvent être réalisées comme suit:

```
struct enreg{  
    int numero ;  
    int qte ;  
    float prix ;  
};  
typedef struct enreg s_enreg;  
s_enreg art1 , art2 ;
```

ou bien

```
typedef struct enreg{  
    int numero ;  
    int qte ;  
    float prix ;  
} s_enreg;  
s_enreg art1 , art2 ;
```

Structure comportant des tableaux

- Soit la déclaration suivante :

```
struct personne {  
    char nom [30];  
    char prenom [30] ;  
    float heures [31];  
};  
struct personne employe, courant ;
```

Cette déclaration réserve les emplacements pour deux structures nommées **employe** et **courant**. Ces dernières comportent trois champs :

- **nom** qui est un tableau de 30 caractères ;
- **prenom** qui est tableau de 30 caractères
- **heures** qui est un tableau de 31 flottants

Structure comportant des tableaux

- ```
struct personne {
 char nom [30];
 char prenom [30] ;
 float heures [31];
};
struct personne employe, courant ;
```
- **employe.heures[4]** désigne le cinquième élément du tableau **heure** de la structure **employe** . il s'agit d'un élément de type *float*.
- **employe.nom[0]** représente le premier caractère du champ **nom** de la structure **employe** .
- **&courant.heures[4]** représente l'adresse du cinquième élément du tableau **heure** de la structure **courant** .
- **courant.nom** représente le champ **nom** de la structure **courant**, c'est-à-dire plus précisément l'adresse de ce tableau

# Tableaux de structures

- Soit la déclaration suivante : nom de modèle de structure

```
struct point {
 char nom;
 int x ;
 int y;
};
```

tableau de 50 éléments du type **point**  
struct point courbe [50] ;

- **courbe[4].nom** représente le **nom** de l'élément de rang **4** du tableau **courbe**.
- **courbe[4].x** désigne la valeur du champ **x** de l'élément de rang **4** du tableau **courbe**.
- **courbe[4]** représente la structure de type point correspondant au cinquième élément du tableau courbe.
- **courbe** est un identificateur de tableau.



# Imbrication de structure

---

- Une structure imbriquée est une structure qui compte parmi ces champs une structure.

```
struct date {
 int jour;
 int mois ;
 int annee;
};
```

```
struct personne {
 char nom [30];
 char prenom [30] ;
 float heures [31];
 struct date date_embauche ;
 struct date date_poste ;
};

struct personne employe, courant ;
```

- **employe.date\_embauche.annee** représente l'année d'embauche correspondant à la structure **employe**. il s'agit d'une valeur de type int.