


Éléments d'informatique – Cours 10. Structures (enregistrements)

Pierre Boudes

30 novembre 2010



This work is licensed under the *Creative Commons Attribution-NonCommercial-ShareAlike 3.0 License*.

- *Éléments d'architecture des ordinateurs (+mini-assembleur)* 
- *Éléments de systèmes d'exploitation*
- Programmation structurée impérative (éléments de langage C)
 - *Structure d'un programme C*
 - *Variables : déclaration (et initialisation), affectation*
 - *Évaluation d'expressions*
 - *Instructions de contrôle : if, for, while*
 - *Types de données : entiers, caractères, réels, tableaux, enregistrements*
 - *Fonctions d'entrées/sorties (scanf/printf)*
 - *Écriture et appel de fonctions*
 - *Débogage*
- *Notions de compilation*
 - *Analyse lexicale, analyse syntaxique, analyse sémantique*
 - *préprocesseur du compilateur C (include, define)*
 - *Édition de lien*
- Algorithmes élémentaires
- Méthodologie de résolution, manipulation sous linux

Plan de la séance

Intérêt des structures (enregistrements)

Déclaration d'un type utilisateur struct

Utilisation d'un type utilisateur struct

Intérêt des structures (enregistrements)

Intérêt des structures (enregistrements)

Un donnée de type structure est composée de plusieurs valeurs, dont le nombre et le type sont fixés à la déclaration de la structure. Par exemple, on pourra définir une structure relevé météo qui

Intérêt des structures :

- *lisibilité* : regrouper un ensemble de données dans un même type, nommé de façon explicite, facilite la relecture du code ;

Intérêt des structures (enregistrements)

Un donnée de type structure est composée de plusieurs valeurs, dont le nombre et le type sont fixés à la déclaration de la structure. Par exemple, on pourra définir une structure relevé météo qui

Intérêt des structures :

- *lisibilité* : regrouper un ensemble de données dans un même type, nommé de façon explicite, facilite la relecture du code ;
- *extension* : les structures permettent d'écrire des fonctions qui retournent plusieurs valeurs, en l'absence de pointeurs.

Intérêt des structures (enregistrements)

Un donnée de type structure est composée de plusieurs valeurs, dont le nombre et le type sont fixés à la déclaration de la structure. Par exemple, on pourra définir une structure relevé météo qui

Intérêt des structures :

- *lisibilité* : regrouper un ensemble de données dans un même type, nommé de façon explicite, facilite la relecture du code ;
- *extension* : les structures permettent d'écrire des fonctions qui retournent plusieurs valeurs, en l'absence de pointeurs.
- *modularité* : on peut rajouter des champs très facilement.

Intérêt des structures (enregistrements)

Un donnée de type structure est composée de plusieurs valeurs, dont le nombre et le type sont fixés à la déclaration de la structure. Par exemple, on pourra définir une structure relevé météo qui

Intérêt des structures :

- *lisibilité* : regrouper un ensemble de données dans un même type, nommé de façon explicite, facilite la relecture du code ;
- *extension* : les structures permettent d'écrire des fonctions qui retournent plusieurs valeurs, en l'absence de pointeurs.
- *modularité* : on peut rajouter des champs très facilement.

On accède aux éléments d'une structure à l'aide de la notation pointée.

Déclaration d'un type utilisateur struct

```
/* Declaration des constantes et types utilisateur */  
struct bulletin_s  
{  
    double temperature;  
    int force;  
};
```

Utilisation d'un type utilisateur struct

```
struct bulletin_s mafonction(struct bulletin_s x);

int main()
{
    struct bulletin_s x = {0.5, 4};
    struct bulletin_s y;

    x.temperature = 13.4;
    y = x;
}
```