

Introduction à la programmation C++

Les structures

Nicolas AUDEBERT

Mercredi 5 octobre 2016



retour sur innovation

Plan de la séance

Avant de commencer

Les structures

Définition

Utilisation

- ▶ **Indenter** : rendre le code lisible, vérifier les ouvertures/fermetures des accolades
- ▶ **Expliciter** : donner des noms et des types explicites aux variables
- ▶ **Débugguer** : vérifier les valeurs prises avec les variables
- ▶ **Lire les messages d'erreur** : toujours retourner à la première erreur
- ▶ **Supprimer les warnings** : supprimer des sources de bugs

C++

```
if(a = 3){...} // ERREUR: le symbole d'egalite est '=='  
if(a == 3){...}
```

C++

```
for(int i; i < 10; i++){...} // ERREUR: il faut initialiser i  
for(int i = 0; i < 10; i++){...} // OK
```

Erreurs avec les tableaux

C++

```
if(a && for(int i = 0; i < 100; i++){tab[i]}){...} // ERREUR

bool test = true;
for(int i = 0; i < 100; i++){
    if(! tab[i])
        test=false;
}
if(a && test){...}
```

C++

```
void f(double tab[8]){...} // argument : un tableau

void g(){
    ...
    double vec[8]; // tableau de 8 cases
    f(vec[8]); // ERREUR : vec[nombre] est le contenu d'une case (un nombre)
               // et en plus, la case 8 n'existe pas !

    f(vec); // OK on appelle la fonction sur la variable vec (un tableau)
}
```

Plan de la séance

Avant de commencer

Les structures

Définition

Utilisation

Pour l'instant...

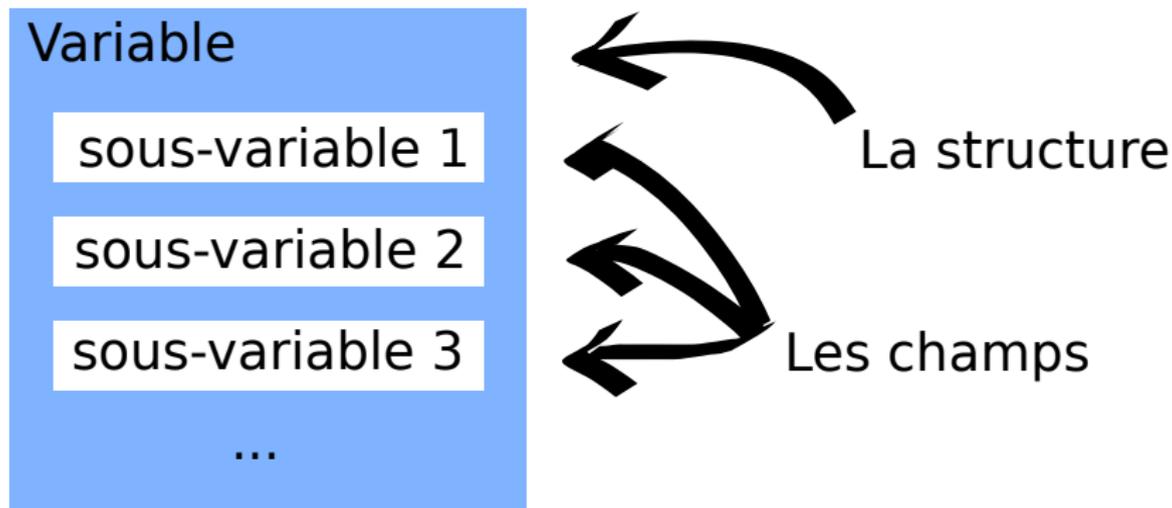
- ▶ factoriser le code : **les fonctions**
- ▶ regrouper les variables de même type : **les tableaux**

Et maintenant...

Regrouper des variables qui ne sont pas forcément du même type mais qui sont cohérentes :

- ▶ Contacts : nom, date de naissance, adresse...
- ▶ Dessin : forme, couleur, épaisseur du trait...
- ▶ ...

On utilise des **structures**.



Les structures définissent de nouveaux types.
Les éléments de la structure sont appelés des champs.

Plan de la séance

Avant de commencer

Les structures

Définition

Utilisation

Cas général

```
struct nom_structure{  
    type1 var1; // les champs  
    type2 var2;  
    ...  
}; // À ne pas oublier
```

La structure définit un nouveau type qui s'utilise comme les autres.
On accède aux champs avec un `.`

Cas général

```
variable_struct.var1 = ...  
cout << variable_struct.var2 << endl;
```

Tableaux dans les structures

Très, très fortement DÉCONSEILLÉ : problèmes liés à l'égalité entre tableaux, notamment dans les retours de fonctions.

Tableaux de structures

Aucun problème, ils se comportent comme des variables classiques.

C++

```
struct Client{
    string nom, prenom;
    string adresse;
    int naissance[3]; // JJ, MM, AAAA
    double taille;
    double poids;
    bool lunettes;
};

...
Client cli1;
cli1.nom = "Presley"
cli1.prenom = "Elvis"
cli1.naissance[0] = 8;
cli1.naissance[1] = 1;
cli1.naissance[2] = 1935;
...
```

C++

```
struct Point{
    double x,y;
};

struct Cercle{
    Point centre;
    double rayon;
    Color couleur;
};

...
Cercle c;
c.centre.x = 0.5;
c.couleur = RED;
Point pt;
pt.x = pt.y = 5.5;
c.centre = pt;

Point p1={1,2}, p2;
p2 = p1 // OK, recopie champ a champ
```

Plan de la séance

Avant de commencer

Les structures

Définition

Utilisation

C++

```
Point pt;  
pt.x = pt.y = 5.5;
```

```
Cercle c;  
c.couleur = RED;  
c.rayon = 3;  
c.centre.x = 5.5;  
c.centre.y = 5.5  
//ou  
c.centre = pt;
```

C++

```
Point pt = {5.5};  
Cercle c = {pt, 3, RED};
```

C++

```
Cercle c={{5.5,5.5},3,RED};
```

L'ordre des éléments pour l'initialisation est l'ordre de définition des champs dans la structure.

Attention

C++

```
Cercle c;  
c = {{5.5,5.5},3,RED}; // ERREUR
```

Les structures fonctionnent comme des types classiques :

▶ Argument

C++

```
void affiche(Point p){  
    cout << p.x << " " << p.y << endl;  
}
```

▶ Retour

C++

```
Point milieu(Point p, Point q){  
    Point m;  
    m.x = (p.x+q.x)/2;  
    m.y = (p.y+q.y)/2;  
    return m;  
}
```

Les structures fonctionnent comme des types classiques :

- ▶ Passage par référence

C++

```
void init(Point &p){  
    p.x = 0;  
    p.y = 0;  
}
```

- ▶ Simulation de système planétaire
- ▶ Système physique
- ▶ Utilisation des structures

N'oubliez pas

INDENTÉZ le code !!

