

Le constructeur par copie

AP2 - programmation objet en C++

Semestre 2, année 2009-2010

Département d'informatique
IUT Bordeaux 1

Février 2010

Vous connaissez déjà...

deux constructeurs

- sans argument (par défaut)
- avec paramètres

exemple

```
class Point
{
public:
    Point();
    Point(float x, float y);
};
```

le constructeur sans argument

Déclaration

```
Point();
```

- sert à **initialiser des variables** et des **éléments de tableaux** avec des **valeurs par défaut**

exemple

```
Point p1, p2;  
Point tab[20];
```

par défaut

Si on ne déclare pas de constructeur sans argument, le constructeur sans argument **par défaut** initialise chaque variable membre avec son constructeur sans argument.

Exemple

```
class Machin {  
    private:  
        int my_n;  
        Truc my_t;  
};  
...  
Machin m;  
// my_n initialisé à 0  
// appel constructeur Truc()
```

le constructeur par copie

déclaration

```
Point (const Point & p);
```

- initialise un objet **en copiant** un autre objet

exemples

```
Point p1 (autrePoint);  
Point p1 = autrePoint; // autre notation
```

et aussi...

- lors d'un **passage de paramètre par copie**

exemple

```
float distance( Point p1, Point p2)
{
    ...
} ;
```

par défaut

Si on ne déclare pas de constructeur de copie :
le constructeur de copie **par défaut** (implicite) initialise chaque variable membre par copie

Exemple

```
class Machin {  
    private:  
        int my_n;  
        Truc my_t;  
};  
...  
Machin m (m2);  
// my_n initialisé à partir de m2.my_n  
// my_t ..... m2.my_t
```