

Introduction
au langage C++ :
C avec des classes

January 23, 2001

Déclaration d'une classe

```
// Fichier: date.h

class CDate {
public:
    // Pour changer la date.
    void    Change(int jour, int mois, annee);

    // Lecture de la date.
    int     Jour() const;    // Note: const
    int     Mois() const;
    int     Annee() const;

    // Passe à la date suivante.
    void    Suivante();

    // Ecrit la date dans un fichier.
    void    Ecrit(FILE*);

private:
    int jour, mois, annee;
};
```

Les classes se manipulent à peu près comme les struct:

```
struct S {  
    /* ... */  
};
```

```
class CDate {  
    /* ... */  
};
```

```
void main()  
{  
    S s;  
    FILE* fichier;  
  
    Cdate date; // "instance" (objet) appartenant  
                // à la classe CDate.  
  
}
```

Implémentation

On utilise la notation

```
Nom_de_la_Classe::Fonction_Membre
```

pour définir une fonction d'une classe.

Les variables déclarées dans une classe donnée sont accessible dans la définition des fonctions membres de la classe.

```
// Exemple:
```

```
void CDate::Ecrit(FILE* f)
{
    printf(f, "%d %d %d" , jour, mois annee);
}
```

```
// Fichier: date.cpp

// Cette fonction change la date.
CDate::Change(int nouv_jour,
              int nouv_mois,
              int nouv_annee)
{
    jour = nouv_jour;
    mois = nouv_mois;
    annee = nouv_annee;
}

// Jour courant.
int CDate::Jour() const
{
    return jour;
}
```

Exemple d'utilisation

```
void main()
{
    CDate date;
    date.Change(3, 1, 2000); // Y2K proof!
    date.Suivante();
    date.Ecrit(stdout);

    date.jour = 4; // Illégal!
                  // Le compilateur va
                  // se plaindre...

    int jour = date.Jour();
}
```

Constructeurs

Dans le programme précédent,

```
CDate date;  
date.Change(3, 1, 2000);
```

c'est un peu long, on peut faire mieux avec un *constructeur*: c'est une fonction membre spéciale qui construit un objet à partir de valeurs initiales.

Syntaxe:

```
// Dans date.h  
class CDate {  
public:  
    CDate();  
    CDate(int jour, int mois, int annee);  
    /* ... */  
};
```

```
// Dans date.cpp,  
// on définit deux fonctions:  
  
CDate::CDate() :  
    jour(0), mois(0), annee(0)  
{  
}  
  
CDate::CDate(int p_jour,  
             int p_mois,  
             int p_annee) :  
    jour(p_jour), mois(p_mois), annee(p_annee)  
{  
}  
  
// Utilisation (dans main()):  
  
CDate date1;  
CDate date2(25,12,1999);
```


Destructeurs

Sert à faire le "ménage" lorsqu'un objet cesse d'exister...

Par convention, cette fonction porte le nom de la classe préfixé du caractère ~

Usuellement, si une allocation de mémoire a été effectuée dans un constructeur, on libère la mémoire dans le destructeur.