

Le programme Hello Word voir **hello.cpp**

Variables en c++ Identique à C pour l'allocation statique

Pointeurs et tableaux Identique à C, voir **echange.c** et **tableau.c**

Références voir **ref.c**, **swapref.c**, **constref.c**

Une référence en C++ est un alias pour une autre variable. On donne un autre nom à une variable qui existe déjà.

On s'en sert essentiellement pour des passages de paramètres.

On utilise aussi les références pour :

- alias d'une variable contenue dans une structure plus complexe
- donnée membre de classe
- valeur de retour de méthode
- désigner un alias constant à une autre variable, en particulier dans le passage de paramètre en entrée.

Fonctions voir **foncmin.c**

Les déclarations et définitions des fonctions suit les principes du C, avec en plus la possibilité de surcharger.

Structures et classes voir **personne.c**

Le C++ offre le moyen de définir des types complexes en agrégeant

les autres types via les struct (éléments publiques) et les class (éléments privés). On peut définir des méthodes pour les classes, des fonctions membres pour les struct.

Déclarations possibles dans le fichier d'entête (.h, .H, .hpp, .hh, .hxx, .h++).

Définitions dans le fichier source (.cpp, .C, .cxx, .cc, .c++, .C++).

On peut aussi tout écrire dans le fichier entête (utile en programmation générique).

Méthodes et fonctions

- Fonction : rien à voir avec une classe
- Méthode de classe
- Méthode d'instance

Rôles particuliers

- Constructeur : appelé sur l'objet juste après son allocation en mémoire
 - Par défaut : sans paramètres
 - Par copie : le paramètre est du même type
- Opérateur d'affection : =
- Destructeur : juste avant désallocation

Intro C++ offre presque tous les mécanismes de POO usuels :

- Méthodes de classe et d'instance
- Attributs de classe et d'instance
- Héritage simple et multiple
- Classes abstraites, interfaces
- Polymorphisme (dynamique)
- Encapsulation
- Transtypage

Mais pour des raisons d'efficacité, on a pas :

- Mécanismes d'introspection. On peut néanmoins tester le type d'un objet à l'exécution.
- On ne peut pas changer le code d'une méthode à l'exécution.

Héritage, sous typage : On peut préciser qu'une classe possède tous les attributs et méthodes d'une autre classe.

Polymorphisme dynamique : En POO, l'héritage sert souvent à manipuler de façon homogène des objets de types distinct (ex : les IHM)

En C++, on indiquera qu'une méthode est polymorphe via le mot clé `virtual`.

Si une méthode est virtuelle, alors le destructeur de la classe doit être virtuel.

On peut forcer une méthode à être abstraite (donc forcément virtuelle) en écrivant `=0` lors de sa déclaration. Dans ce cas la classe est abstraite et ne peut être instanciée.