



Language de programmation C++

C'est quoi ?

Le C++ est un langage de programmation de haut niveau, développé par Bjarne Stroustrup au début des années 1980.

Il est une extension du langage C et offre des fonctionnalités de programmation orientée objet, de programmation générique et de programmation fonctionnelle, en plus des capacités de programmation procédurale du C. C++ est utilisé pour développer des logiciels systèmes, des jeux vidéo, des applications de bureau, et des systèmes embarqués, entre autres. Il est apprécié pour sa performance, sa flexibilité et son efficacité.

Principe de fonctionnement : Exemple TV

Définir une Classe

Créer des instances de classes = objets

Attributs

Marque, Volume, Chaine ... Les attributs sont privés (encapsulés) non visible par l'utilisateur

Méthodes
(Public)

SetColor(), ChangeChaine(int, nb).. Fonction permettant de modifier les attributs

Projets réalisés (en autonomie)

1) Initiation aux bases de C++ (IDE : Visual Studio Code 2022)

- Création d'un fichier source. **main.cpp**
- Initialisations des variables. **type name = .. ;**
- Les différents types de variables. **int, char, float, double, bool**
- Opérations mathématiques. **+ - * / % |**

```
1 //auteur : Enzo MONDOVY
2
3 // #include = Intruction/directive de préprocesseur
4 // <iostream> = header de la bibliothèque standard qui permet d'utiliser les flux
5 #include <iostream>
6 int main()
7 {
8     // type name;
9     // char - character - 'r'
10    // int - integer - 50,42, -400
11    // float - floating points - 4.2f
12    // double - double precision - 5.0
13
14    // Declaration + Initialization
15
16    int a = 50;
17    int b = 40;
18    int c = a + b + 10 - 2 * 3;
19
20    // std:: = utilisation de l'espace de nom de la bibliothèque standard, pour pouvoir accéder à toutes
21    // les fonctions qui sont cachées dans l'espace de nom (encapsulation pour éviter les conflits de noms
22    // entre bibliothèques
23    std::cout << a+b << std::endl; //Ouverture d'un flux de sortie sur la console dans lequel est er
24    //une chaîne de caractères "...
25    std::cin.ignore(); //Bloquer l'exécution du programme à cette ligne
26
27
28    return 0;
29 }
```

```
1 //auteur : Enzo MONDOVY
2 int main()
3 {
4     // type name;
5     // char - character - 'r'
6     // int - integer - 50,42, -400
7     // float - floating points - 4.2f
8     // double - double precision - 5.0
9
10    // Declaration + Initialization
11    int age = 42;
12    float mark = 18.5f;
13    double height = 200.50;
14
15    //Operators : + * - / % |
16    int a = 50;
17    int b = 40;
18    int c = a + b + 10 - 2 * 3;
19    float r = 10.0f * 30.0f;
20
21
22    return 0;
23 }
```

2) Communication avec la console

- Fichier d'en tête `#include`
- Affichage de texte en console : `std::cout << " texte " << std::endl ;`
- Saisie Utilisateur `std::cin >>[valeurs a saisir] ;`
- Utilisation de la console de commande

Algorithme de demande et affichage de l'âge :

```
Main.cpp* x
Tutorial_01 (Global Scope) main()
1 //autor : Enzo MONDOVY
2
3 // #include = Intruction/directive de préprocesseur
4 // <iostream> = header de la bibliothèque standard qui permet d'utiliser les flux
5 #include <iostream>
6 int main()
7 {
8     int age = 0;
9     std::cout << "How old are you" << std::endl; // affichage de texte en console
10    std::cin >> age; //> age = récupération de la saisie utilisateur et stockage dans la variable age.
11    std::cin.ignore(); //cin.ignore permet au flux de supprimer ce qui se trouve mémoire courte
12
13
14    std::cout << "Your are : " << age << std::endl;
15    std::cin.ignore();
16
17    //std:: permet de faire appel à un fonction native de C++
18    //cin = console input
19    //cout = console output
20    //endl = endlime = pour faire un saut de ligne
21
22    return 0;
23 }
```

Algorithme qui sert de calculatrice (uniquement pour l'addition de 2 entiers) :

```
1 //autor : Enzo MONDOVY
2
3 // #include = Intruction/directive de préprocesseur
4 // <iostream> = header de la bibliothèque standard qui permet d'utiliser les flux
5 #include <iostream>
6 int main()
7 {
8     int a = 0, b = 0;
9
10    std::cout << "Enter a then b : "; //message en console
11    std::cin >> a >> b; //saisie utilisateur a et b séparer d'un espace
12    std::cin.ignore();
13
14    std::cout << "a + b = " << a + b << std::endl; //afficher le resultat de a+b
15    std::cin.ignore();
16
17    return 0;
18 }
```

3) Algorithmes sous conditions

- Syntaxe : `if(condition)`

```
Main.cpp* [X]
Tutorial_01 (Global Scope) main()
1 //author : Enzo MONDOVY
2
3 #include <iostream>
4 int main()
5 {
6     bool condition = false; //true or false
7
8     if (condition)           //si la condition est vérifiée alors faire le programme entre {}
9     {
10        //....
11    }
12    else                       //si la condition n'est pas vérifiée alors faire le programme entre {}
13    {
14        //....
15    }
16    return 0;
17 }
```

- Attribution de récompenses en fonction de la moyenne :

```
//author : Enzo MONDOVY
#include <iostream>
int main()
{
    float average = 0;
    std::cin >> average;
    std::cin.ignore();

    if (average == 20)
    {
        std::cout << "Felicitation!" << std::endl;
    }
    else if (average >= 16)
    {
        std::cout << "Très bien!" << std::endl;
    }
    else if (average >= 14)
    {
        std::cout << "Bien!" << std::endl;
    }
    else if (average >= 12)
    {
        std::cout << "Assez bien!" << std::endl;
    }
    else
    {
        std::cout << "Pas de mention" << std::endl;
    }
    std::cin.ignore();
    return 0;
}
```

(Cas pratique d'un jeu vidéo)

- Jeux vidéo : définir si un joueur est en vie en fonction de son hp

```
//author : Enzo MONDOVY
#include <iostream>
int main()
{
    int hp = 5;

    if (hp > 0) //opérateurs de comparaison ==, !=, >=, <=, <, >, && (et), || (ou)
    {
        std::cout << "The player is alive." << std::endl;
    }
    else
    {
        std::cout << "The player is dead" << std::endl;
    }
    std::cin.ignore();
    return 0;
}
```

- Filtrage d'accès avec saisie utilisateur

```
//author : Enzo MONDOVY
#include <iostream>
int main()
{
    int age = 0;
    bool with_parents = false;
    std::cout << "How old are you?" << std::endl;
    std::cin >> age;
    std::cin.ignore();

    if (age >= 16 || with_parents)
    {
        std::cout << "Autorise" << std::endl;
    }
    else
    {
        std::cout << "Not Autorise" << std::endl;
    }

    std::cin.ignore();
    return 0;
}
```

4) Algorithmes sous conditions + saisie utilisateur

- En fonction de l'âge saisi par l'utilisateur en déduire de sa majorité.

```
//author : Enzo MONDOVY
#include <iostream>
int main()
{
    int age = 0;
    std::cout << "How old are you ?" << std::endl;
    std::cin >> age;
    std::cin.ignore();

    if (age >= 18)
    {
        std::cout << "You are of age" << std::endl;
    }
    else
    {
        std::cout << "You are underage" << std::endl;
    }
    std::cin.ignore();
    return 0;
}
```

- Version améliorée

```
//author : Enzo MONDOVY
#include <iostream>
int main()
{
    int age = 0;
    std::cout << "How old are you ?" << std::endl;
    std::cin >> age;
    std::cin.ignore();

    //Inline-if (opérateur, prenant en argument 3 paramètres)
    std::cout << "You are " << (age < 18 ? "underage" : "over 18") << std::endl;

    std::cin.ignore();
    return 0;
}
```

5) Boucles

- **For (int i = 0 ; N ; ++i)**

Saisie utilisateur permettant l'affichage des nombres de 1 à N :

```
//author : Enzo MONDOVY
#include <iostream>
int main()
{
    //1 à 10
    //for (for loop)

    int N = 0;

    std::cout << "Add N value" << std::endl;
    std::cin >> N;
    std::cin.ignore();

    for (int i = 0; i <= N; ++i) //for (variant de boucle; condition d'exécution de la boucle, pas')
    {
        //afficher les nombre de 0 à N
        std::cout << i << std::endl;
    }

    std::cin.ignore();
    return 0;
}
```

Suite géométrique :

```
//author : Enzo MONDOVY
#include <iostream>
int main()
{
    int a = 4;
    int r = 3;

    int N = 0;
    std::cin >> N;
    std::cin.ignore();

    for (int i = 0; i < N; ++i)
    {
        a *= r;
    }

    std::cout << a << std::endl;
    std::cin.ignore();

    return 0;
}
```