# Cours 0 — Installation de C++ sur Windows et macOS

## Objectifs pédagogiques

- Comprendre le rôle d’un compilateur C++

- Installer et configurer un environnement de développement C++ sous Windows et macOS

- Vérifier l’installation et compiler un premier programme simple

- Travailler avec un éditeur de code comme Visual Studio Code

## Introduction

C++ est un langage de programmation polyvalent, orienté objet, très utilisé dans le développement d’applications lourdes, de jeux vidéo, de logiciels système et dans le domaine de la recherche.  
  
Avant de pouvoir exécuter du code C++, il faut un compilateur, c’est-à-dire un programme qui traduit le code source en langage machine. Sur Windows, on utilise généralement MinGW (version Windows du compilateur GCC). Sur macOS, c’est Clang, inclus dans les outils de développement Xcode.

## I. Installation de C++ sous Windows

### 1. Télécharger MinGW

MinGW signifie Minimalist GNU for Windows. C’est un ensemble d’outils permettant de compiler du code C et C++ sur Windows.  
  
Lien de téléchargement : https://osdn.net/projects/mingw/releases/  
Téléchargez le fichier : mingw-get-setup.exe

### 2. Installer MinGW

1. Lancez le fichier téléchargé.  
2. Sélectionnez les paquets suivants :  
 - mingw32-gcc-g++ (compilateur C++)  
 - mingw32-gcc-objc  
 - msys-base  
3. Allez dans Installation → Apply Changes pour lancer l’installation.

### 3. Configurer la variable d’environnement PATH

Cette étape permet d’utiliser g++ depuis n’importe quel dossier dans la ligne de commande.  
1. Ouvrez : Panneau de configuration → Système → Paramètres système avancés → Variables d’environnement.  
2. Dans la section Variables système, recherchez Path puis cliquez sur Modifier.  
3. Ajoutez la ligne suivante : C:\MinGW\bin  
4. Cliquez sur OK pour enregistrer les modifications.

### 4. Vérifier l’installation

Ouvrez l’invite de commande (cmd) et tapez :  
g++ --version  
  
Si une version s’affiche, votre installation est réussie.

## II. Installation de C++ sur macOS

### 1. Installer les outils de développement Xcode

macOS inclut un compilateur via les Command Line Tools.  
  
Ouvrez le terminal (Cmd + Espace → Terminal) puis tapez :  
xcode-select --install  
  
Acceptez l’installation et attendez la fin du processus.

### 2. Vérifier l’installation

Dans le terminal :  
clang++ --version  
  
Si une version s’affiche (exemple : Apple clang version 15.x.x), le compilateur est installé correctement.

### 3. (Optionnel) Installer un IDE moderne

Pour un environnement de travail plus confortable :  
- Téléchargez Visual Studio Code : https://code.visualstudio.com/  
- Installez l’extension C/C++ (éditeur Microsoft)  
- Créez un fichier main.cpp et un dossier pour vos projets.

## III. Test de fonctionnement

Créez un fichier nommé main.cpp :  
  
#include <iostream>  
using namespace std;  
  
int main() {  
 cout << "Bonjour, C++ fonctionne !" << endl;  
 return 0;  
}

Sous Windows :  
g++ main.cpp -o main  
main.exe  
  
Sous macOS :  
clang++ main.cpp -o main  
./main  
  
Résultat attendu : Bonjour, C++ fonctionne !

## IV. Exercices pratiques

Exercice 1 — Vérification du compilateur

- Exécutez g++ --version ou clang++ --version  
- Notez le nom et la version de votre compilateur

Exercice 2 — Programme de test

- Créez et compilez le code précédent  
- Modifiez-le pour afficher votre nom et votre système d’exploitation

Exercice 3 — Observation d’erreurs

- Supprimez accidentellement un point-virgule ou une accolade  
- Observez le message d’erreur du compilateur  
- Corrigez l’erreur et recompilez

## Résumé du cours

|  |  |  |
| --- | --- | --- |
| Élément | Windows | macOS |
| Compilateur | MinGW / g++ | Clang / clang++ |
| Vérification | g++ --version | clang++ --version |
| IDE recommandé | VS Code, Code::Blocks | VS Code, Xcode |
| Commande de compilation | g++ main.cpp -o main | clang++ main.cpp -o main |

## Conclusion

Vous avez désormais un environnement prêt à l’emploi pour développer en C++. Vous avez également appris à compiler et exécuter un programme simple. Dans le prochain cours, nous étudierons la structure d’un programme C++, les types de données et les instructions de base.