

INFO2009 - Introduction à l'informatique

Comment installer et utiliser GCC

Bertrand Alexis

Université de Liège

2026

Contact

Mail

alexis.bertrand@uliege.be

Bureau

1.81a

Institut Montefiore (b28)

Table des matières

Installation de GCC

Linux

MacOS

Windows

Écrire son premier programme

Compiler avec GCC et exécuter son programme

Pour tout problème d'installation ou des questions, tu peux m'envoyer un mail à alexis.bertrand@uliege.be



Avant de commencer

Avant de commencer à installer quoi que ce soit, vérifie que GCC n'est pas déjà présent sur ta machine :

Ouvre un terminal et tape `gcc --version`. Si la commande te retourne la version actuelle de GCC (ou de Clang), tu peux déjà compiler tes codes C. Sinon, tu peux trouver dans ce tutoriel comment installer GCC en fonction de ton système d'exploitation.

Ouvrir un terminal

- ▶ Windows : Le terminal peut être lancé en allant dans la barre de recherche de Windows et en entrant *cmd*.
- ▶ MacOS : Le terminal se trouve dans tes *Applications*, il se situe initialement dans le dossier *Utilities* (fr: *Utilitaires*).
- ▶ Linux : Le terminal est accessible en effectuant le raccourci clavier *Ctrl+Alt+T*.

Installer GCC sur Linux (1/2)

Attention

Certaines distributions utilisent d'autres package managers que celui utilisé pour ce tutoriel. Si tu as une distribution qui n'utilise pas apt, n'hésite pas à venir me trouver !

Installer GCC

- ▶ Ouvre un terminal grâce aux touches **Ctrl+Alt+T** ou en cherchant dans tes applications
- ▶ Tape les commandes suivantes :
 - ▶ `sudo apt update && sudo apt upgrade`
 - ▶ `sudo apt install build-essential`



Installer GCC sur Linux (2/2)

GCC devrait avoir été installé avec le package *build-essential*.

- Vérifie que GCC est correctement installé en faisant :
`gcc --version`

Tu peux désormais utiliser les commandes GCC pour compiler ton code C !

Installer Clang sur MacOS (1/2)

Pour pouvoir installer notre compilateur C, nous allons passer par l'outil de ligne de commande de MacOS qui contient de nombreux paquets tel que Clang.

Installer l'outil de ligne de commande

- ▶ Ouvre un terminal (tu peux le trouver dans tes applications ou avec Spotlight)
- ▶ Tape la commande : `xcode-select --install`

Aide

Tu peux facilement trouver quelque chose sur ta machine grâce à *Spotlight* avec les touches **Cmd+Espace**.

Installer Clang sur MacOS (2/2)

Lors de l'exécution de la dernière instruction, Clang devrait avoir été installé.

- ▶ Vérifie que Clang est correctement installé en faisant :
`gcc --version`

Tu peux désormais utiliser les commandes GCC ou Clang pour compiler ton code C !

Remarque

Ici, nous avons installé Clang et non GCC, qui est un autre compilateur. Cependant, MacOS crée un lien entre les commandes GCC et Clang, ce qui permet de compiler avec Clang avec les mêmes instructions que GCC.

Installer GCC sur Windows

Avant d'installer MSYS2

L'installation de *MSYS2* dépend du type d'architecture de ta machine (x86-64 ou ARM64). Assure-toi de l'architecture de ta machine avant de continuer. Tu peux vérifier l'architecture de ta machine en allant dans :

- ▶ Information système
- ▶ Dans la colonne Type du système

Installer GCC sur Windows x86-64 (1/3)

Installer MSYS2

Rends-toi sur le site de MSYS2 et télécharge l'installateur x86-64 avant de l'exécuter.

Terminal dans l'environnement de MSYS2

Une fois *MSYS2* installé, tu peux lancer le terminal de l'environnement *UCRT64* dans lequel tu installeras le compilateur et au travers duquel tu pourras compiler tes codes. Tu devrais facilement le trouver via le menu Démarrer en écrivant *MSYS2 UCRT64*.

Installer GCC sur Windows x86-64 (2/3)

Mettre MSYS2 à jour

Avant de continuer quoi que ce soit, il faut mettre l'environnement à jour. Pour ceci, exécute la commande suivante dans le terminal *UCRT64* :

► `pacman -Suy`

MSYS2 demandera de fermer le terminal afin de terminer la mise à jour.

Installer GCC sur Windows x86-64 (3/3)

Installer GCC

Rouvre le terminal et exécute la commande suivante qui te permettra d'installer gcc :

- ▶ `pacman -S mingw-w64-ucrt-x86_64-gcc`

gcc devrait maintenant être installé.

- ▶ Vérifie que gcc est correctement installé en faisant `gcc --version`

Installer Clang sur Windows ARM64 (1/4)

Il n'y a pas de paquet pour gcc sur l'architecture arm. Nous devons donc installer Clang pour ensuite créer un lien entre les commandes gcc et Clang afin de pouvoir utiliser les commandes gcc pour exécuter les commandes du compilateur Clang, exactement comme pour MacOS.

Installer MSYS2

Rends-toi sur le site de MSYS2 et télécharge l'installateur ARM64 avant de l'exécuter.

Installer Clang sur Windows ARM64 (2/4)

Terminal dans l'environnement de MSYS2

Une fois MSYS2 installé, tu peux lancer le terminal de l'environnement *CLANGARM64* dans lequel tu installeras le compilateur et au travers duquel tu pourras compiler tes codes. Tu devrais facilement le trouver via le menu Démarrer en écrivant `MSYS2 CLANGARM64`.

Installer Clang sur Windows ARM64 (3/4)

Mettre MSYS2 à jour

Avant de continuer quoi que ce soit, il faut mettre l'environnement à jour. Pour ceci, exécute la commande suivante dans le terminal *CLANGARM64* :

▶ `pacman -Suy`

MSYS2 demandera de fermer le terminal afin de terminer la mise à jour.

Installer Clang sur Windows ARM64 (4/4)

Installer Clang

Rouvre le terminal et exécute la commande suivante qui te permettra d'installer Clang :

- ▶ `pacman -S mingw-w64-clang-aarch64-clang`

Maintenant nous allons installer un deuxième paquet permettant de lier les commandes gcc avec le compilateur Clang :

- ▶ `pacman -S mingw-w64-clang-aarch64-gcc-compat`

Les commandes gcc devraient maintenant être reconnues.

- ▶ Vérifie que le compilateur est correctement installé en faisant `gcc --version`

Éditeur de texte et IDE (1/2)

Afin d'écrire ton premier programme en C, il te faudra un éditeur de texte. Les fichiers que tu écriras ont une extension particulière (.c) donc les programmes spécifiques à certaines extensions comme Word ou GoogleDoc ne permettent pas d'écrire un fichier compilable.

Éditeur de texte

Il y a des dizaines d'éditeurs de texte et tu peux même utiliser l'éditeur de texte propre à ton OS comme *Notes* ou *TextEdit* pour écrire ton code.

Un éditeur de texte disponible pour tous les OS avec couleur syntaxique est par exemple gedit ou Sublime Text.

Éditeur de texte et IDE (2/2)

IDE

Les IDE (Environnement de Développement Intégré) sont des applications qui regroupent l'éditeur de texte, le débogueur, le compilateur et d'autres outils pour faciliter l'écriture de code. Nous recommandons de ne pas utiliser ce genre d'outil dans le cadre de ce cours afin que vous vous familiarisiez avec la compilation et que vous n'ayez pas d'outils facilitant l'écriture de votre code, aide que vous n'aurez pas avec vous lors de l'examen. Un IDE disponible pour tous les OS est par exemple Visual Studio Code.

Parcourir les dossiers et fichiers (1/3)

Lorsque tu veux compiler un fichier que tu as créé, il faut que ton terminal soit ouvert dans le dossier dans lequel le fichier se trouve.

Pour ce faire, il va falloir naviguer par ton terminal dans tes dossiers. Tu peux faire ceci grâce à la commande `cd nom_du_sous-dossier`. `cd` veut dire "Change Directory" et permet donc de passer d'un dossier à un autre jusqu'à arriver au dossier dans lequel ton fichier se cache. Si jamais tu es allé trop loin, tu peux utiliser la commande `cd ..` afin de revenir au dossier précédent.

Exemple

```
cd Documents/Introduction_informatique
```

Parcourir les dossiers et fichiers (2/3)

Si tu ne sais pas où tu te situes, tu peux voir le chemin absolu du dossier (c'est-à dire la succession de dossiers dans lesquels tu es rentré pour y accéder) dans lequel tu es grâce à la commande `pwd`. Si tu ne sais pas quels fichiers ou dossiers il y a dans le dossier dans lequel tu te situes pour le moment, la commande `ls` va lister tous les fichiers et dossiers dans ce répertoire.

Exemple

`ls`

- ▶ `main.c`
- ▶ `autre_dossier`

Il te suffira alors de naviguer à travers tes dossiers afin de compiler ton code C.

Parcourir les dossiers et fichiers (3/3)

Si tu ne parviens pas à entrer dans le bon dossier. Tu peux trouver le chemin absolu du dossier via ton explorateur de fichier et alors effectuer la commande

- ▶ `cd /<chemin>` (où `<chemin>` est le chemin absolu récupéré)

pour te retrouver directement dans ce dossier. Attention que sur Windows, le chemin récupéré contiendra des `\`. Il faudra les remplacer par des `/`.

Compiler un fichier

Quand tu as écrit ton code, avant de pouvoir tester ton programme, tu dois d'abord le compiler de cette manière

Commande pour compiler

```
gcc -Wall -Wextra nom_fichier.c -o nom_fichier_exécutable
```

Bonne pratique

Il est recommandé de toujours ajouter `-Wall` et `Wextra` lors de la compilation d'un code C. Ajouter ces flags permet de facilement repérer les erreurs de compilations et les warnings !

Exemple

```
gcc -Wall -Wextra main.c -o HelloWorld
```

Lancer un fichier exécutable

Une fois que tu as compilé ton ou tes fichiers C en un fichier exécutable, tu peux lancer ton programme de cette façon

Exécuter un fichier

```
./fichier_exécutable
```

Exemple

```
./HelloWorld
```