

Que faire en cas de problème lors de la compilation des codes d'exemples ?

Nombreux codes donnés dans ce cours. Il peut arriver que vous ayez des problèmes de compilation. La raison est que le C++ n'est pas unique, il existe en fait plusieurs versions du C++ (normes) que l'on nomme C++03, C++11, C++14 et C++1z (il existe aussi un C++98, mais il est tellement ancien que tous les compilateurs actuels prennent en charge au moins le C++03 - sauf si vous travaillez sur un environnement particulier).

Selon le compilateur que vous utilisez, toutes ces normes ne sont pas forcément prises en charge et pas forcément en totalité.

Les codes sont écrit en C++14 et ont été testé sur coliru.com, donc avec le compilateur Clang version 3.5. Si cette version de Clang ne prend pas en charge une fonctionnalité, cela est indiqué dans le texte. Si vous utiliser un autre compilateur, il faudra vérifier que ce n'est pas un problème de prise en charge du C++14 sur votre compilateur.

Vous verrez par la suite comment installer une chaîne de compilation complète, quelques remarques ici sur les compilateurs.

GCC et MingW

Compilateur en général fournit avec les IDE sous windows, par exemple Code::Block et Qt Creator. MingW est un portage de GCC sous windows, mais a une version de retard (GCC est en version 4.9, MingW en version 4.8.2). MingW ne supporte donc pas les dernières fonctionnalités du C++11 et encore moins le C++14. Utiliser une version de GCC pour windows ([lien](#))

De plus, par défaut, nouvelles normes non activée. Il faut le faire manuellement :

- ligne de commande = `-std=c++1z`, `-std=c++14`, `-std=c++1y`, `-std=c++11`, `-std=c++03` ;
- dans Qt Creator : CONFIG += C++11 dans le .pro
- dans Code::Block : cocher l'option "activer le C++11" ou ajouter une directive de compilation (C++14)

Microsoft Visual

Prendre le dernier MSVC 2013 CRTP2. Pas d'option à activer.

Clang

Compilateur par défaut sur Mac (XCode). Installable dans les paquets sous linux. Problème sous windows (portage en cours)

Chapitre précédent	Sommaire principal	Chapitre suivant
------------------------------------	------------------------------------	----------------------------------

[Cours, C++](#)