

Software specifications

Chapter number	Software required (With version)	Free/ Proprietary	If proprietary , can code testing be performed using a trial version	If proprietary, then cost of the software	Download links to the software	Hardware specifications	OS required
All	GCC, Clang, Microsoft Visual C++ Compiler, or Intel C++ compiler. Versions must be recent enough to support C++17.	Free: GCC, Clang. Proprietary : Microsoft Compiler, Intel C++ compiler	Microsoft VS and Intel Parallel Studio provide 30/60 day trial versions.	Free: GCC, Clang. Monthly/Yearly license fee: Microsoft Visual Studio and Intel Parallel Studio	GCC: https://gcc.gnu.org/releases.html Clang: http://releases.llvm.org/download.html Microsoft Visual Studio: http://www.visualstudio.com/ Intel Parallel Studio XE 2017: https://software.intel.com/en-us/intel-parallel-studio-xe/try-buy	GCC, Clang: 1 GB of RAM and 20GB of disk. A Virtual Machine with this characteristics should suffice. Microsoft Visual Studio 2017: 1.8 GHz or faster processor, 4GB of RAM, and approx. 1GB hard disk space.	GCC, Clang: GNU/Linux (Any distribution), MacOS (Any Version), Windows (Any version). Microsoft Visual Studio 2017: Windows (Windows 7 SP1, 8.1, 10), Mac OS 10.11 or newer. Intel Parallel Studio: Windows (Windows 7, 8.1, 10 64bit), Linux (any distribution)

Detailed installation steps (software-wise)

The homepage of the Standard C++ Foundation provides an article for getting started with C++ that is regularly maintained and lists popular compilers and development environments: <https://isocpp.org/get-started>.

Windows:

There are multiple choices:

Microsoft Visual Studio comes directly from Microsoft and is the standard choice on Windows. Install simply by downloading it and following the instructions of the installation wizard: <https://www.visualstudio.com/>

Intel Parallel Studio XE 2017 provides the C++ compiler from Intel for Computers with Intel chipsets. Download and install from <https://software.intel.com/en-us/intel-parallel-studio-xe/try-buy>.

There are also possibilities to install only *open source* toolchains:

MinGW stands for “Minimalist GNU for Windows” and provides a minimalist development environment for native Microsoft Windows applications using an open source toolchain including GCC. Install by following the instructions on http://www.mingw.org/wiki/Getting_Started

Cygwin is a kind of package manager which allows for maintaining a Unix-like environment on Windows. It provides packages for GCC and Clang. In order to install a full development environment, it is recommended to install the whole package category `Devel`: <https://cygwin.com/install.html>

Mac OS X:

clang is the default compiler on Mac OS and is included in Xcode, which can be installed directly from the App Store, or downloaded from <https://developer.apple.com/xcode/>.

In order to install more recent versions of `clang` and `g++`, you can use the free and open source **Homebrew** package manager.

1. Follow the installation instructions for Homebrew on <http://brew.sh>
2. Install `g++` via `$ brew install gcc`

3. Install clang via `$ brew install llvm`

You can check which versions are available using `$ brew search <package name>`.

Ubuntu Linux

You can just install g++ and clang via the standard package manager:

```
$ sudo apt install build-essential g++ clang
```

Fedora Linux

You can just install g++ and clang via the standard package manager:

```
$ sudo dnf install gcc-c++ clang
```

Online compilers

With the following online compilers it is possible to enter, compile, and run C++ code on the internet into a text field in the browser.

- <http://wandbox.org>
- <http://coliru.stacked-crooked.com>
- <https://godbolt.org> (Doesn't run the code, but shows the generated assembly)