0.6 — Installing an Integrated Development Environment (IDE)

👤 **ALEX** 🕐 **JULY 28, 2021**

An Integrated Development Environment (IDE) is a piece of software that contains all of the things you need to develop, compile, link, and debug your programs.

With a typical C++ IDE, you get a code editor that does line numbering and syntax highlighting. Many (but not all) IDEs include a C++ compiler and a linker, which the IDE will know how to interface with in order to convert your source code into an executable file. And when you need to debug your program, you can use the integrated debugger.

Furthermore, IDEs typically bundle a number of other helpful editing features, such as integrated help, name completion, auto-formatting, and sometimes a version control system. So while you could do all of these things separately, it's much easier to install an IDE and have them all accessible from a single interface.

So let's install one! The obvious next question is, "which one?". Many IDEs are free (in price), and you can install multiple IDEs if you wish, so there's no "wrong decision" to be made here. We'll recommend a few of our favorites below.

If you have some other IDE in mind, that's fine too. The concepts we show you in these tutorials should generally work for any decent modern IDE. However, various IDEs use different names, layouts, key mappings, etc... so you may have to do a bit of searching in your IDE to find the equivalent functionality.

Tip

To get the most value of this tutorial, we recommend installing an IDE that comes with a C++17 capable compiler.

If you're restricted to using a compiler that only supports C++14 or C++11 (due to educational or business constraints), most of the lessons and examples will still work. However, if you encounter a lesson that uses concepts from C++17 (or newer) and you're using an older language compiler, you'll have to skip it or translate it to your version, which may or may not be easy.

You should not be using any compiler that does not support at least C++11 (which is typically considered the modern minimum spec for C++).

We recommend installing the newest version of a compiler. If you can't use the newest version, these are the absolute minimum compiler versions with C++17 support:

- GCC/G++ 7
- clang++ 8
- Visual Studio 2017 15.7

Visual Studio (for Windows)

If you are developing on a Windows machine (as most of you are) and disk space and download size are not a constraint, then we strongly recommend Visual Studio 2019 Community (ignore the Visual Studio 2022 Preview). When you run the installer, you'll eventually come to a screen that asks you what workload you'd like to install. Choose *Desktop development with C++*. If you do not do this, then C++ capabilities will not be available.

The default options selected on the right side of the screen should be fine, but please ensure that the *Windows 10 SDK* is selected. The Windows 10 SDK can be used on older versions of Windows, so don't worry if you're still running Windows 7 or 8.
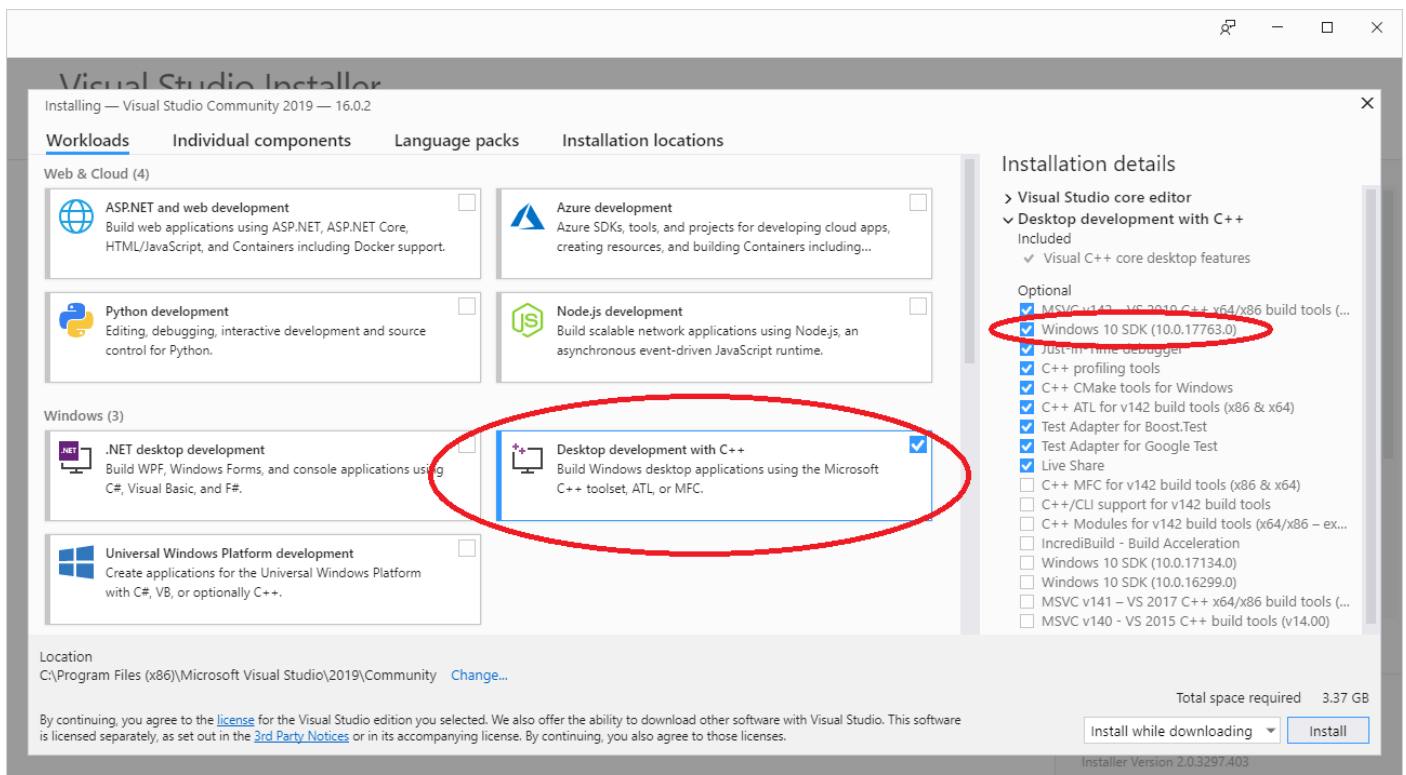
If disk space and/or download size are a challenge, then we recommend Microsoft's free Visual Studio Express 2017 for Windows

Code::Blocks (for Linux or Windows)

If you are developing on Linux (or you are developing on Windows but want to write programs that you can easily port to Linux), we recommend **Code::Blocks**. Code::Blocks is a free, open source, cross-platform IDE that will run on both Linux and Windows.

For Windows users

Make sure to get the version of Code::Blocks that has MinGW bundled (it should be the one whose filename ends in *mingw-setup.exe*). This will install MinGW, which includes a Windows port of the GCC C++ compiler:

Windows XP / Vista / 7 / 8.x / 10:

| File	Date	Download from
codeblocks-20.03-setup.exe	29 Mar 2020	**FossHUB** or **Sourceforge.net**
codeblocks-20.03-setup-nonadmin.exe	29 Mar 2020	**FossHUB** or **Sourceforge.net**
codeblocks-20.03-nosetup.zip	29 Mar 2020	**FossHUB** or **Sourceforge.net**
codeblocks-20.03mingw-setup.exe	29 Mar 2020	**FossHUB** or **Sourceforge.net**
codeblocks-20.03mingw-nosetup.zip	29 Mar 2020	**FossHUB** or **Sourceforge.net**
codeblocks-20.03-32bit-setup.exe	02 Apr 2020	**FossHUB** or **Sourceforge.net**
codeblocks-20.03-32bit-setup-nonadmin.exe	02 Apr 2020	**FossHUB** or **Sourceforge.net**
codeblocks-20.03-32bit-nosetup.zip	02 Apr 2020	**FossHUB** or **Sourceforge.net**
codeblocks-20.03mingw-32bit-setup.exe	02 Apr 2020	**FossHUB** or **Sourceforge.net**
codeblocks-20.03mingw-32bit-nosetup.zip	02 Apr 2020	**FossHUB** or **Sourceforge.net**

For Linux users

Some Linux installations may be missing dependencies needed to run or compile programs with Code::Blocks.

Debian-based Linux users (such as those on Mint or Ubuntu) may need to install the *build-essential* package. To do so from the terminal command line, type: `sudo apt-get install build-essential`.

Arch Linux users may need to install the *base-devel* package.

Users on other Linux variants will need to determine what their equivalent package manager and packages are.

When you launch Code::Blocks for the first time, you may get a *Compilers auto-detection* dialog. If you do, make sure *GNU GCC Compiler* is set as the default compiler and then select the *OK* button.
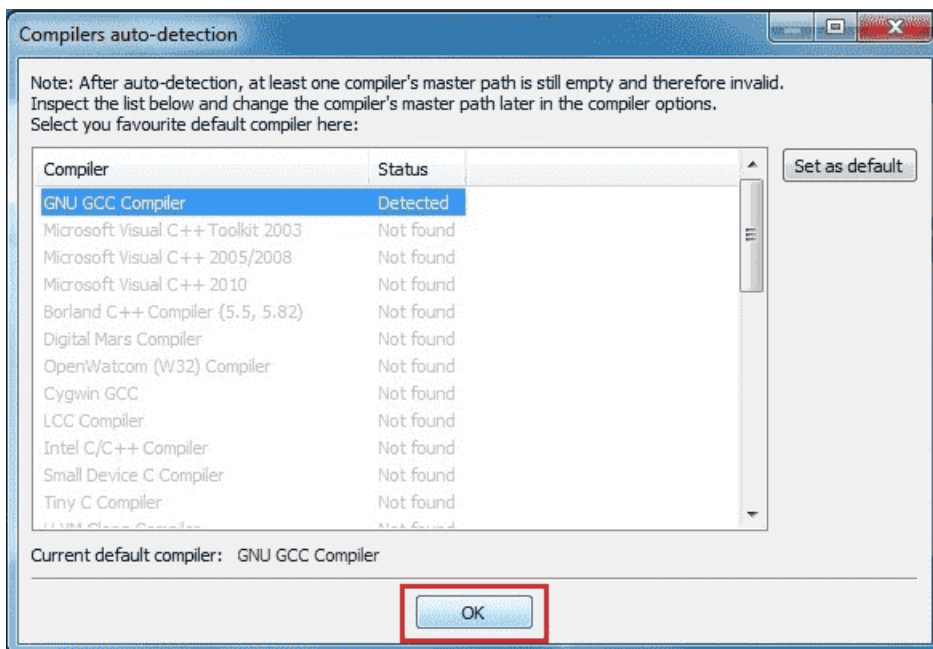
Q: What do I do if I get a "Can't find compiler executable in your configured search paths for GNU GCC Compiler" error?

Try the following:

1. If you're on Windows, make sure you've downloaded the version of Code::Blocks WITH MinGW. It's the one with "mingw" in the name.
2. Try going to settings, compiler, and choose "reset to defaults".
3. Try going to settings, compiler, toolchain executables tab, and make sure "Compiler's installation directory" is set to the MinGW directory (e.g. C:\Program Files (x86)\CodeBlocks\MinGW).
4. Try doing a full uninstall, then reinstall.
5. **Try a different compiler.**

Mac OSX IDEs

Popular Mac choices include **Xcode** (if it is available to you), or **Eclipse**. Eclipse is not set up to use C++ by default, and you will need to install the optional C++ components.

Although Visual Studio for Mac has been released, as of Aug 2018 it does not support C++, so at this time we can not recommend it.

Other compilers or platforms

Q: Can I use a web-based compiler?

Yes, for some things. While your IDE is downloading (or if you're not sure you want to commit to installing one yet), you can continue this tutorial using a web-based compiler, such as the one at **TutorialsPoint**.

Web-based compilers are fine for dabbling and simple exercises. However, they are generally quite limited in functionality -- many won't allow you to save projects, create executables, or effectively debug your programs. You'll want to migrate to a full IDE when you can.

Q: Can I use a command-line compiler (e.g. g++ on Linux)?

Yes, but we don't recommend it for beginners. You'll need to find your own editor and look up how to use it elsewhere. Using a command line debugger is not as easy as an integrated debugger, and will make debugging your programs more difficult.

Q: Can I use other code editors or IDEs, such as Eclipse, Sublime, Notepad++, or Visual Studio Code?

Yes, but we don't recommend it for beginners. There are many great code editors and IDEs that can be configured to support a wide variety of languages, and allow you to mix and match plugins to customize your experience however you like. However, many of these editors and IDEs require additional configuration to compile C++ programs, and there's a lot that can go wrong during that process. For beginners, we recommend something that works out of the box, so you can spend more time learning to code and less time trying to figure out why your code editor isn't working properly with your compiler or debugger.

IDEs to avoid

You should avoid the following IDEs altogether because they do not support at least C++11, do not support C++ at all, or are no longer actively supported or maintained:

- **Borland Turbo C++ -- does not support C++11**
- **Visual Studio for Mac -- does not support C++**

There is no good reason to use an outdated or unsupported compiler when lightweight, free alternatives that support modern C++ exist.

When things go wrong (a.k.a. when IDE stands for "I don't even...")

IDE installations seem to cause their fair share of problems. Installation might fail outright (or installation might work but the IDE will have problems when you try to use it due to a configuration issue). If you encounter such issues, try uninstalling the IDE (if it installed in the first place), reboot your machine, disable your antivirus or anti-malware temporarily, and try the installation again.

If you're still encountering issues at this point, you have two options. The easier option is to try a different IDE. The other option is to fix the problem. Unfortunately, the causes of installation and configuration errors are varied and specific to the IDE software itself, and we're unable to effectively advise on how to resolve such issues. In this case, we recommend copying the error message or problem you are having into a Google search and trying to find a forum post elsewhere from some poor soul who has inevitably encountered the same issue. Often there will be suggestions on things you can try to remedy the issue.

Moving on

Once your IDE is installed (which can be one of the hardest steps if things don't go as expected), or if you're temporarily proceeding with a web-based compiler, you are ready to write your first program!

Leave a comment... Put C++ code between triple-backticks (markdown style):```Your C++ co

Name*

Email*

Notify me about replies: 🔔 **POST COMMENT**

619 COMMENTS

Newest ▾

Ⓧ

Ⓧ