How to install the GNU Compiler Collection (GCC) for Windows

This guide is for installing a GCC compiler in a windows environment. If you are using a Linux machine, you should already have the appropriate tools installed by default. If you are running an OSX machine, you need to refer to the appropriate documentation on installing the GCC compiler using XCode. For our windows environment we will be installing the MinGW (Minimalist GNU for Windows) compiler suite. MinGW is a windows port of the GCC compiler and provides all of the functionality we need for compiling using the CLI and makefiles.

If you are interested in using an IDE for programming in Windows, I recommend either:

- Code::Blocks - http://www.codeblocks.org/
- Microsoft Visual Studio (Community Edition) - https://visualstudio.microsoft.com/

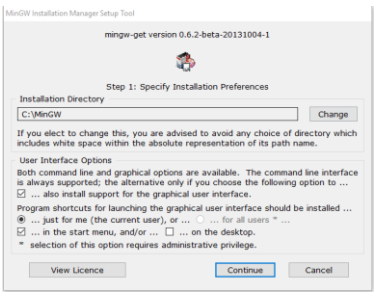
Because we want you to use the GCC compiler for this class, you should use Code::Blocks and be sure to select the MinGW or GNU GCC option for compiler. You can use Visual Studio, but you will need to configure it to use the GCC compiler instead of the Visual C++ compiler that comes with Visual Studio.

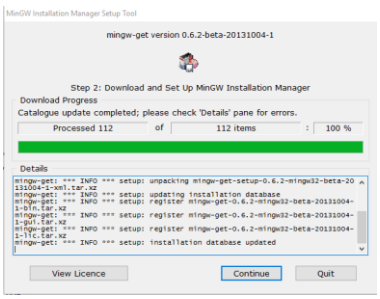
Note: Even if you are using an IDE, you will still need to make sure your source code compile correctly using GCC. You will still need to create makefiles that work with GCC as well, or we will not be able to check your work.

If you are feeling brave and do not need an IDE for programming and want to code in a basic text editor, then you should check out:

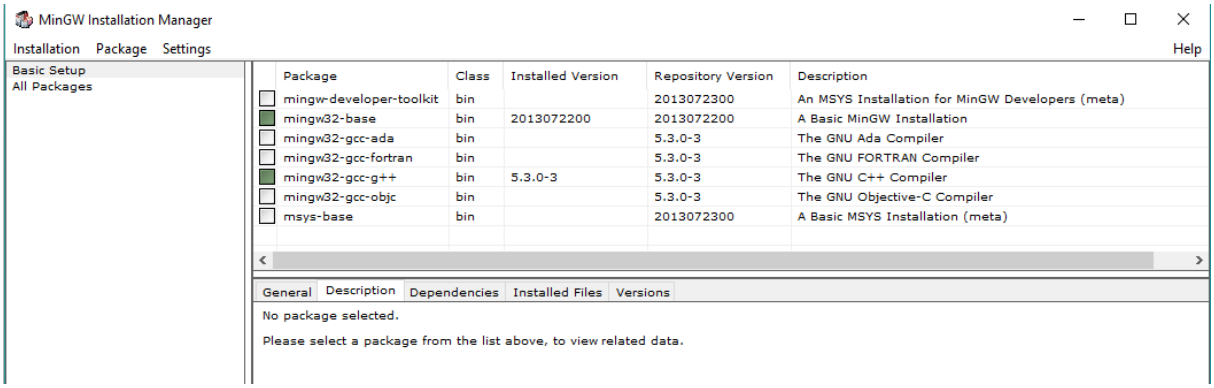
- Sublime Text - http://www.sublimetext.com/2 (Free, but asks you to purchase)
- Atom - https://atom.io/ (Free, and open-source, so you can create your own functionality)
- Vim/Emacs/gedit – (Should be installed by default in Linux OS)

1. **Download and install the GCC compiler suite**
   https://sourceforge.net/projects/mingw/files/latest/download
   Go to the link above and click on the download link for “mingw-get-setup.exe” and run it.





   Make sure to remember where you are installing MinGW because you need to make sure the MinGW/bin directory is listed in your PATH (instructions below). Setup your preferences and wait for the installer to finish downloading packages. Click continue once finished.



You should now see the installation manager screen. Make sure you click to install the **mingw32-base** and **mingw32-gcc-g++** packages. On the top menu, click installation -> Apply Changes. Close the installer once it is finished.

2. **Finish up configuring GCC.**
   Now that you have successfully installed MinGW, you need to include the directory of executables in your system path so that you can call them from the command line. I personally recommend using Path Editor - https://patheditor2.codeplex.com/ . It is a very easy to use tool for adding and removing directories to your path.
   If you are not using path editor in windows, then go to
   a. System settings (Right click on your start menu icon and select system or search for it).
   b. Click advanced system settings.
   c. Under the advanced tab, click environment variables.
   d. Under system variables, click Add and then give the title and location of your MinGW/bin folder location (The default location is *C:/MinGW/bin*). Click ok when finished.

3. **IMPORTANT: Setup make.exe**
   Your GCC installation also contains the executable make.exe that we use to automate the compilation process using makefiles. However, the make.exe in the MinGW directory is mislabeled. Follow these steps to fix it:
   a. Navigate to your MinGW installation directory. (E.g. *C:/MinGW/bin*)
   b. Find the file **mingw32-make.exe,** make a copy of it and rename it to **make.exe**
   c. You now should be able to open a new terminal and verify that you can now call the make and GCC compile functions by typing **gcc** and **make**. (You should see errors relating to not supplying the commands with arguments. If you see an error about not recognizing the "gcc" or "make" command, then you need to go back and make sure you added the correct directory to your system path.

   To compile a program named yourProgram.c into an executable named yourProgram.exe, you need to type:
   - gcc –o yourProgram yourProgram.c
   - yourProgram (type ./yourProgram if you are running linux)

   That's it, now you are ready to start writing and compiling C programs for the class!