

CS1342 - Lab 01

This assignment is due **Sunday, Jan. 31 @ 11:59pm**.

In this lab, you will be setting up your machine for the ability to build, execute and submit programs for this class. This lab has 2 parts to the setup:

1. Setting up your machine (C++ Compiler and IDE)
2. Setting up Git / Github account.

Once you have completed the assignment, please upload a screenshot of a CLion hello world output (example provided below), along with your GitHub username for full credit. It is important that you follow these instructions carefully to ensure correct setup.

If any problems arise with your machine make sure to communicate that with your professor as soon as possible to ensure ample time to get it corrected.

Part 1 - Setting up GCC/G++

Follow the instructions below based on your operating system on your laptop or personal computer that you will use for this class. For both operating systems you will be installing a compiler (gcc/g++) and installing CLion, the IDE of choice for this course.

MacOS

If you have a windows machine, please continue to the Windows section of this lab.

GCC/G++ Setup

To install GCC/G++ compilers for C++ we must first install a package manager that will help us manage downloads. We will be using homebrew to do so for this class.

Homebrew Installation

1. Open up the `Terminal` application. (You can search for it in the spotlight).
2. Once a new terminal is open, paste the following line in the terminal and then press `Enter`

```
/usr/bin/ruby -e "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

3. To verify installation run `brew --version` to check the version of homebrew you have installed

GCC/G++ Installation

Run the following command in Terminal

```
brew install gcc
```

Note: This can take some time to install and compile all the dependencies. This may take up to an hour on some machines

Once this command completes you may move onto CLion Setup. If you run into errors take a screenshot and post in Slack!

CLion Setup

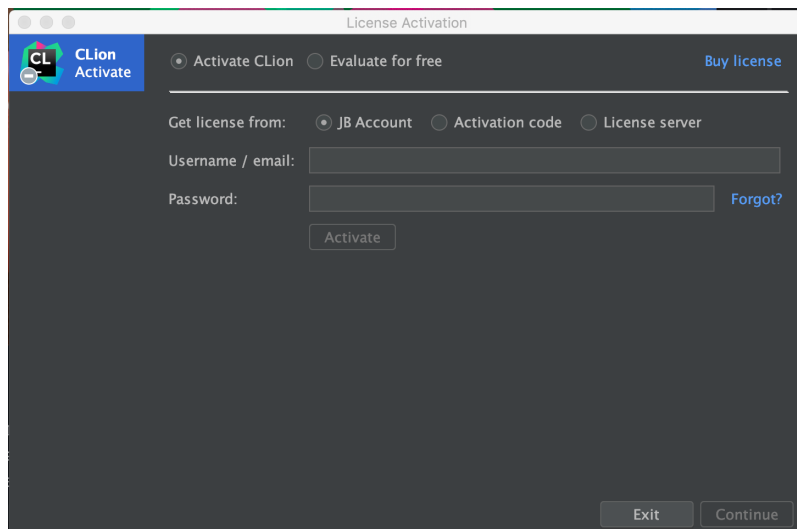
This section will walk you through the installation and setup of the CLion IDE that we will be using for all programming assignments this semester.

Prerequisites

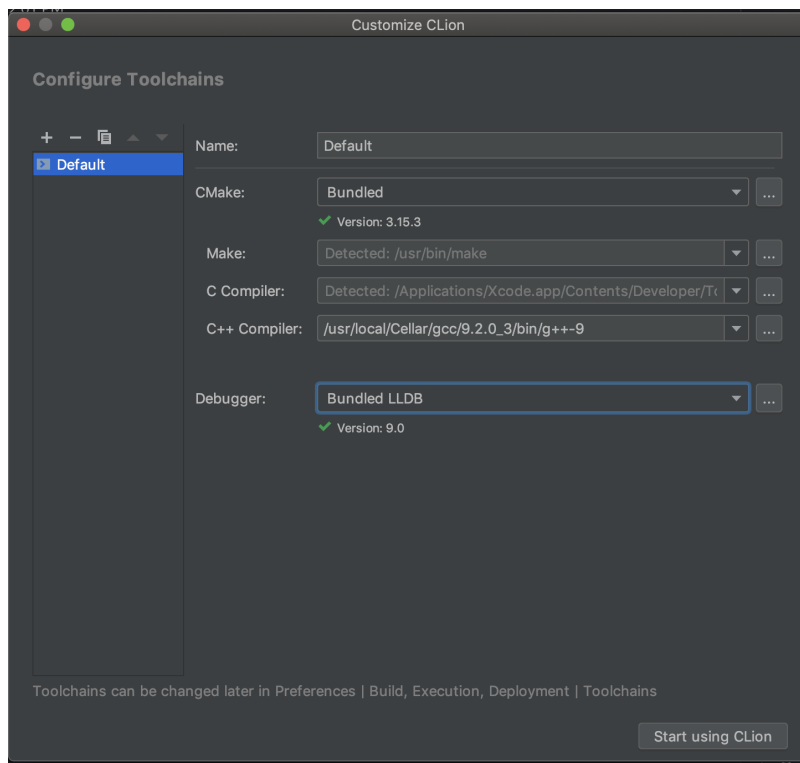
You have a student license for JetBrains Products allowing you to download CLion for free. If you do not have a license please visit <https://www.jetbrains.com/shop/eform/students> and provide your smu email for a free license.

Installation

1. Once you have a Student License you may download CLion by visiting the downloads page <https://www.jetbrains.com/clion/download/>. Then click `Download`.
2. Double click the downloaded Clion setup file and walk through the setup. You may choose all the default settings.
3. Next you will have to activate your CLion account. To do so make sure `Get license from` has `JB Account` selected, then enter your username and password that you created for your student license.



4. Click `Next` after your license has been verified.
5. Configure Toolchains - CLion will automatically configure your application to use the default XCode clang compiler. We want to configure it to use the gcc/g++ compiler we just installed using homebrew. To do so locate the `g++-9` executable (it should be located here `/usr/local/Cellar/gcc/9.2.0_3/bin/g++-9`). Then paste the full path (`/usr/local/Cellar/gcc/9.2.0_3/bin/g++-9`) into the `C++ Compiler` field. *Note: Your gcc version number may be different. Ensure you copy the correct version number*



6. Select `Start using CLion` to complete the CLion setup!

7. If you would like to test your installation, create a new CMake project (This will create a new C++ Hello World Program), then run the project (It should print out 'Hello World!'). To run a program in CLion you can select the green arrow in the top right corner of the IDE.

8. After successfully running your first program, continue on to Part 2!

NOTE: For all of your programs you will not need to create a new project. You will be opening up a starter project that will be given to you through GitHub. You will be walking through this process in LabQuiz 2

Windows Setup

This tutorial will walk you through the setup process for getting the gcc/g++ compiler installed on windows with a unix toolchain and the CLion IDE provided by JetBrains. This document will walk you through 2 steps:

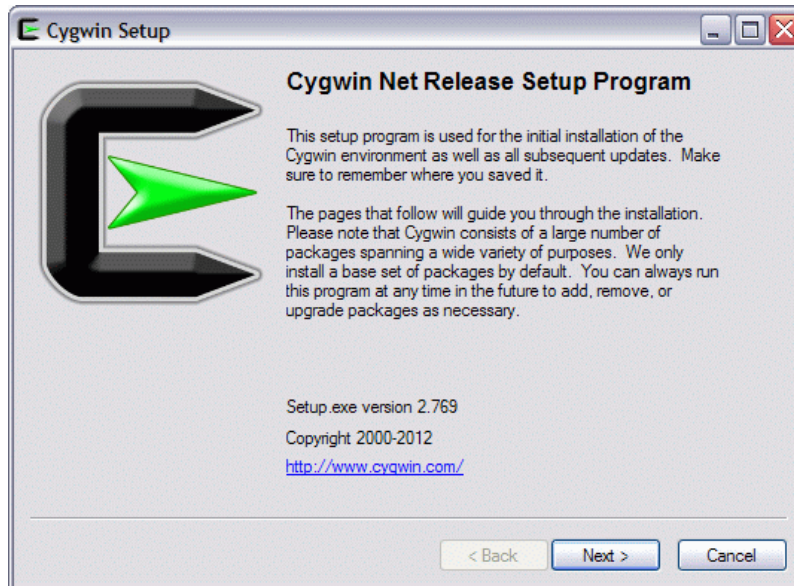
1. Installing a Unix/Linux Tool for windows with a c++ compiler.
2. Installing and running your first program on CLion.

GCC/G++ Setup

A large majority of C Libraries and Packages are written for the Unix environment. Windows is not a Unix Environment. Cygwin and MinGW are toolchains that help get those packages and libraries to run in a Windows machine. CygWin/MinGW creates the Unix look and feel on the Windows environment. You may choose either CygWin or MinGW for your linux environment. This tutorial will walk you through the setup with CygWin. (For MinGW Setup see the **More Info** section.)

CygWin Installation

1. First, download CygWin by visiting cygwin.com/install.html
2. Choose the correct setup script depending on your operating system (either `setup-x86-64.exe` or `setup-x86-32.exe`). If you do not know which one to choose it is most likely that you will need the 64 bit executable as most modern computers are run using 64 bit OS.
3. After you have downloaded the setup script, double click it to open up the installer window.



4. Use the recommended settings and continue through by clicking `next` until you get to the **Installation Location**.
5. Choose Installation Directory. It is usually recommended that you just use your `C:` drive. So your location will look something like this: `C:\cygwin` .
6. Choose Local Package Directory. This will be the location where CygWin installs other C/C++ libraries and tools for programming. You may leave it in the downloads folder if you like or change it to `C:\Storage\Cygwin` .
7. Follow the recommended settings until you get the the **Select Packages** screen.
8. Select the following packages:
 - o `gcc-core`
 - o `gcc-g++`
 - o `libgcc1`
 - o `gdb`
 - o `make`
 - o `libmpfr4` (Note: this may already be selected on some systems. If already selected you can ignore this library)

Using the search bar, you can type in the name of each package, then select the latest version from the `Skip` dropdown (Do not select any `Test` versions).

9. Continue and wait for packages to download (This may take some time, feel free to grab some coffee!), then select `Finish` .

You should now be able to open up a new CygWin terminal window and run `g++ --version` from the command line and see the current g++ version appear in the console. If you see it then you have successfully installed a C++ compiler! Next move onto **CLion Setup**

More information on CygWin or MinGW

For more information on CygWin or MinGW you may visit <https://www.jetbrains.com/help/clion/quick-tutorial-on-configuring-clion-on-windows.html>. This is a quick tutorial on getting a windows environment configured for CLion.

CLion Setup

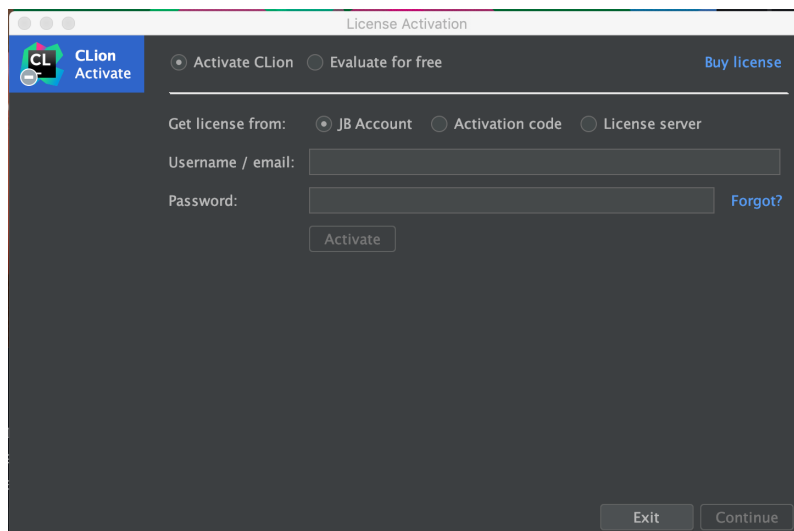
This section will walk you through the installation and setup of the CLion IDE that we will be using for all programming assignments this semester.

Prerequisites

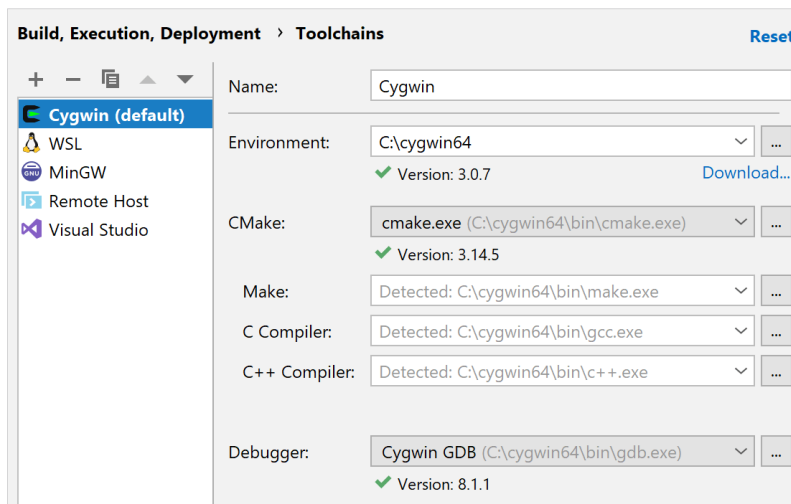
You have a student license for JetBrains Products allowing you to download CLion for free. If you do not have a license please visit <https://www.jetbrains.com/shop/eform/students> and provide your smu email for a free license.

Installation

1. Once you have a Student License you may download CLion by visiting the downloads page <https://www.jetbrains.com/clion/download/>. Then click `Download`.
2. Double click the downloaded Clion setup file and walk through the setup. You may choose all the default settings.
3. Next you will have to activate your CLion account. To do so make sure `Get license from` has `JB Account` selected, then enter your username and password that you created for your student license.



4. Click `Next` after your license has been verified.
5. To setup CLion you will need to tell it what toolchain to use for compiling your programs. Select Cygwin from the Environment list. CLion will attempt to detect the Cygwin installation automatically. Check the detection result, and specify the path manually if needed.



6. Wait until the tools detection completes, then click `Start using CLion` to complete the CLion setup!. Thats it!

7. If you would like to test your installation, create a new CMake project (This will create a new C++ Hello World Program), then run the project (It should print out 'Hello World!'). To run a program in CLion you can select the green arrow in the top right corner of the IDE.

8. After succesfully running your first program, continue on to Part 2!

NOTE: For all of your programs you will not need to create a new project. You will be opening up a starter project that will be given to you through GitHub. You will be walking through this process in LabQuiz 2

Part 2 - Git / Github Account Setup

What is GitHub?

Github is a software development tool that allows engineers to host their code and collaborate online. You can think of GitHub as a cloud repository that stores copies of your code.

What is Git?

Git is a distributed version-control system for tracking changes in source code during software development. It is designed for coordinating work among programmers, but it can be used to track changes in any set of files. For your programs in this class you will use git frequently to help you work iteratively on your program and submit your source code to GitHub.

For more resources and tutorials on Git and Github I recommend looking through the tutorials here <https://try.github.io/>. While we will not be using Git and Github to the fullest in this class, if you plan to continue in Software development, having prior knowledge of version control with git/github is a huge plus!

GitHub Setup

1. Create your GitHub Account

Sign up for GitHub **using your SMU EMAIL** (@smu.edu) at <https://github.com/join>. You will be using this account for future assignments so make sure you remember your credentials!

2. Download GitHub Desktop

Next download GitHub Desktop which is an application that will help you communicate and manage your local repositories and GitHub. To download GitHub Desktop visit <https://desktop.github.com/>. There are downloads available for both Mac and Windows.

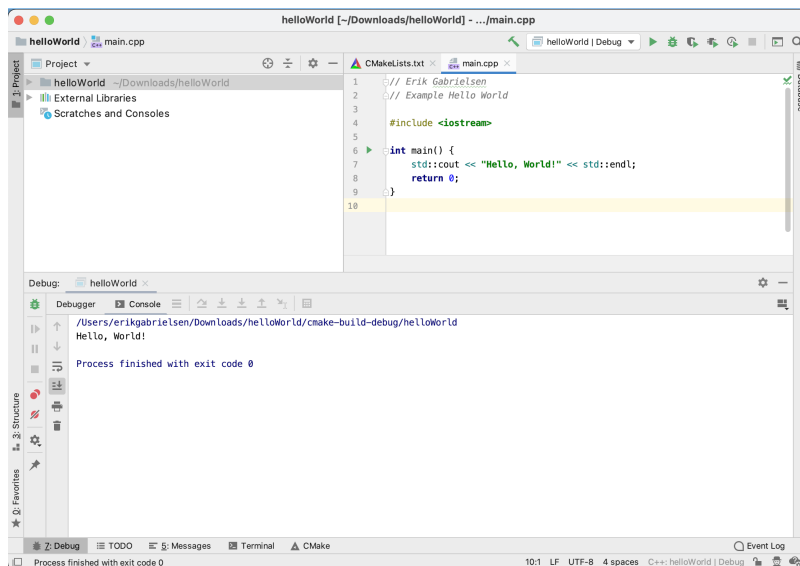
Once you have downloaded GitHub Desktop, sign in using the credentials you used for your GitHub Account in the previous step.

You will be able to use GitHub Desktop for downloading and submitting your programming assignments. (More on this later). For now ensure that you are able to login to the GitHub Desktop Application using your SMU Email.

3. Submit the following for Lab01:

Submit your github username (that you created in part 1 step 1) to Canvas along with a screenshot of the output of your hello world program in CLion for full credit on this assignment by **11:59pm January 31**.

Example output:



Git CLI Setup (Optional)

After installing GitHub Desktop you should have git installed on your machine. To verify open up a Terminal window (MacOS) or Powershell/Cygwin shell (Windows) and type in the following:

```
git --version
```

If you do not get a "Command Not Found" Error message than it has been successfully installed. If you are unsure you may find more instructions for installing git on your machine at <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

From here on out you may choose to use the GitHub Desktop GUI to manage your programs or the git CLI (Command Line Interface). The git CLI allows you to interface with github through the command line or terminal as apposed to an actual GUI. This my preferred method however the learning curve can be tough so if you are more comfortable using the GUI that is fine too.