CSCI 1320 Computer Science I: Engineering Applications – Fall 2018
Instructor: Zagrodzki
Lab 6
Due Friday, October 26, by 6pm

Getting started with Jupyter Hub

As discussed in lecture, the two things needed in order to begin creating programs in C++ (or any programming language) is some kind of a development environment. While MATLAB provided us with a very nice Integrated Development Environment, for C++ we will use two separate components as a development environment within Jupyter Hub:

1. Text editor - for creating source files.
2. Compiler - for generating executable programs from the source code.

The purpose of this exercise is to familiarize ourselves with Jupyter Hub as a development environment for C++. We will guide you to develop the basic tools to get around in the Terminal environment by using Bash commands (Bash is the language used within Terminal). Bash is not described in either of the textbooks required for this class, but the basic commands have/will be covered in lecture. Bash is also heavily documented online (we won't be requiring you to write entire Bash scripts in this class, so don't go down a rabbit hole learning about those for now).

In a text document (~.txt) write out the steps to accomplish the following tasks in your Jupyter Hub environment Any command line commands you will use in the Terminal need to be *spelled out exactly.* For any non-command line steps just describe what you would do (e.g. "Right-click on myFile.zip and click Download").

1. When you open up the Terminal it starts at the root directory. Create a new directory here for keeping your C++ coding (in case you want to use JH for other things in the future, let's create a directory specifically for C++ coding.) Call it something like myCpp or cppProjects.
2. Within this directory, create a new text file. Make this file be a C++ cheat sheet where you will record snippets of syntax to use as a reference as you learn C++.
3. Now create an new directory within this directory for lab 6 and assignment 6.
4. Create a file called hello.cpp.
5. Actually, delete the file from step 4.
6. Create a file called <lastName>Lab6.cpp (e.g. zagrodzkiLab6.cpp).
7. Do quick check to make sure only the one file you expect exists in the current directory.
8. Now open your file from step 6 so that you can edit it (do this *outside* of Terminal - just double click the newly created file under the "Files" tab.) Copy the code from the attached myNameIs.cpp file. Only change the file such that your name is displayed.
9. Compile the file and give the executable some custom name.
10. Run the executable program from step 9.

11. Create a zip file only containing your source code file.
12. Download the zip file.

Submitting the assignment:
Make sure your text file includes your name, student ID, course number, lab number and recitation section. Submit only the single text file on Moodle as Lab 6 by due date.