CPSC256 Development Environment Setup

Disclaimer: All of this should work on MacOS, Windows, or pretty much any modern Linux distribution (Fedora, Centos, Debian, AlmaLinux, Ubuntu for sure).

Step 1: GitHub.com Setup

Note: We are NOT using the PCSE GitLab installation, as you may have used in CPSC 150/250 or some of your other courses. GitHub.com is an alternate solution that we will use in CPSC 256.

Account Setup (if you do not already have a GitHub account)

1. Go to www.github.com
2. Click on "Sign Up"
3. Choose a username (can be anything), use your CNU email address, and choose a password. Verify and create your account.
4. Verify your email address (the email will be sent to you by GitHub).
5. Complete the profile questions

CPSC256 Fork Repository Setup

1. In your GitHub account, in a browser, in the Search bar in the upper left corner, search for "owner:brash99". This will take you to my GitHub repo pages.
2. Click on the Repositories tab
3. Find the repository called CLionProjects, and click on that
4. Near the upper right corner of the page, click on "Fork". Accept the default choices, and then click on the green "Create Fork" button to create your own fork of my CLionProjects repository.
5. Now, you have a copy (fork) or my CLionProjects repository, inside of which you will find a cpsc256 folder that contains all of the example codes that we will go through in class, along with a Documents folder and a LectureNotes folder. See below for how to keep this up-to-date!

Configuring GitHub.com Security

1. In your GitHub account, in a browser, go the Settings, by clicking on your profile picture in the upper right corner, and choosing "Settings".
2. In the long menu shown on the left side, choose "Developer Settings", which is all the way at the bottom of the list of options.

3. Choose “Personal access tokens” -> Tokens(classic) from the new menu on the left.
4. Choose Generate new token -> Generate new token (classic)
5. In the “Note” field, enter cpsc256
6. Choose “No expiration” for the Expiration time
7. Select ALL of the radio buttons in the “Select scopes” region
8. Click on “Generate token”
9. This will generate a new token that looks like ‘ghp_******’. Copy this token to the clipboard!!! This is important, as you are going to need this in the next steps!
10. On your local computer, open up a simple text file, and copy this token into that file. Save the file, giving it some name of your choice.

Step 2: Download and Install CLion

https://www.jetbrains.com/clion/download

Follow the normal installation instructions for the downloaded file.

Step 3: Open, Configure, and Test CLion

Step 3a: Clone your GitHub.com forked repository, created above

1. In the “Welcome to CLion” window, click on: Get from VCS. Or alternatively, you can choose Git->Clone from the top-level menubar.
2. In the URL area, enter the location of your GitHub.com repo … the format is as follows: https://edwardbrash:ghp_UZJZ2GUZORidXcg3wAfsUP1tUbCRt0Hdt5@github.com/edwardbrash/CLionProjects.git

Where you need to:

- Replace ‘edwardbrash’ with YOUR GitHub.com username, in both places!
- Replace ghp_UZJZ2GUZORidXcg3wAfsUP1tUbCRt0Hdt5’ with YOUR Personal access token

3. In the “Directory” area, make sure you are cloning into ~/CLionProjects (where ~ means your home directory)

4. Choose to Trust this repository, and choose to trust all repositories in PyCharmProjects, when asked

5. CLion should open up a “Open Project Wizard” window, where you can choose the native compiler and other options. Just accept the “bundled” versions and hit OK.

Step 3b: Test the “Hello World” program

1. Highlight “CLionProjects” in the Project window (left side of the main window)
2. Expand the cpsc256 folder.
3. Expand the HelloWorld folder.
4. Double-click on main.c
5. You should see that this new file has been opened in the editor window.
6. There should be a message that the ‘Project is not configured’ - choose to ‘Configure CMake Project’
7. Choose the CMakeLists.txt configuration file that is in the ~/CLionProjects/cpsc256 folder
8. You should now see that this and a few other subprojects have been configured!
9. In the top area of the CLion Window, choose the drop-down menu that says “ArtistAndArtwork’, and choose instead ‘HelloWorld’
10. Click on the green right-facing triangle (to RUN the program!)

Step 3c: Keeping your fork up-to-date

The following will pull all changes from my GitHub.com CLionProjects repository into your forked copy:

1. Choose Git -> GitHub -> Sync Fork
2. The first time that you do this, you will have to authorize PyCharm (a JetBrains app) to access your GitHub account. Just follow the instructions when this happens, and once you have provided this authorization, it should work seamlessly going forward.