Page Installing and Using The Xcode IDE

Xcode is free from the App Store on your Macbook. Note that in the rest of this document there are screenshots to help you. Sometimes these are just a portion of the entire application, in order to save space.

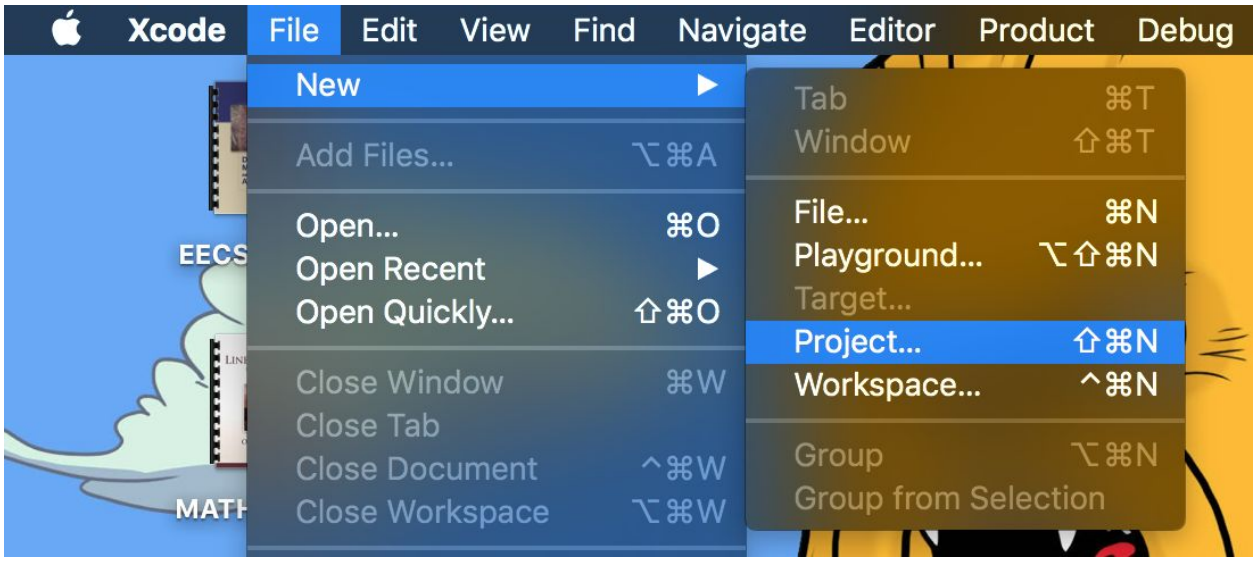
WARNING: Xcode has one known error, in that the size of a deque is not displayed correctly inside the debugger.

Downloading/Installing

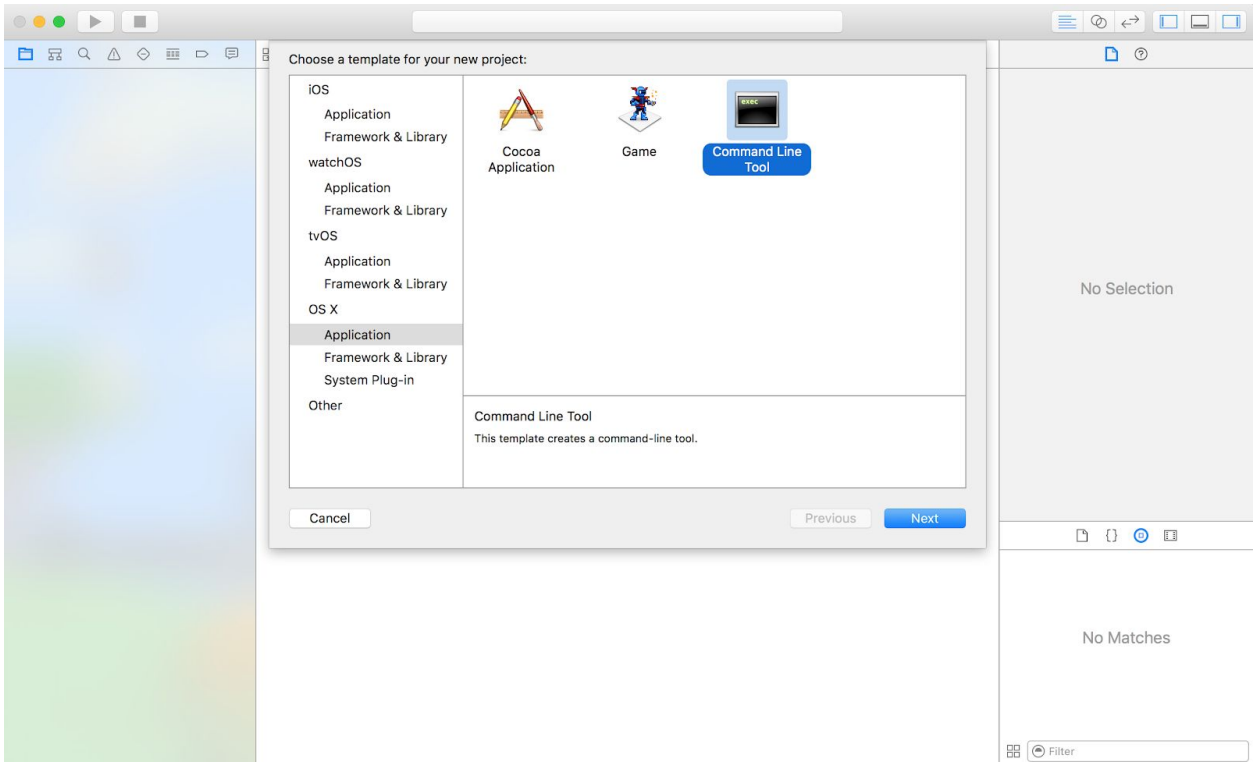
Get Xcode from the App Store. And that's it!

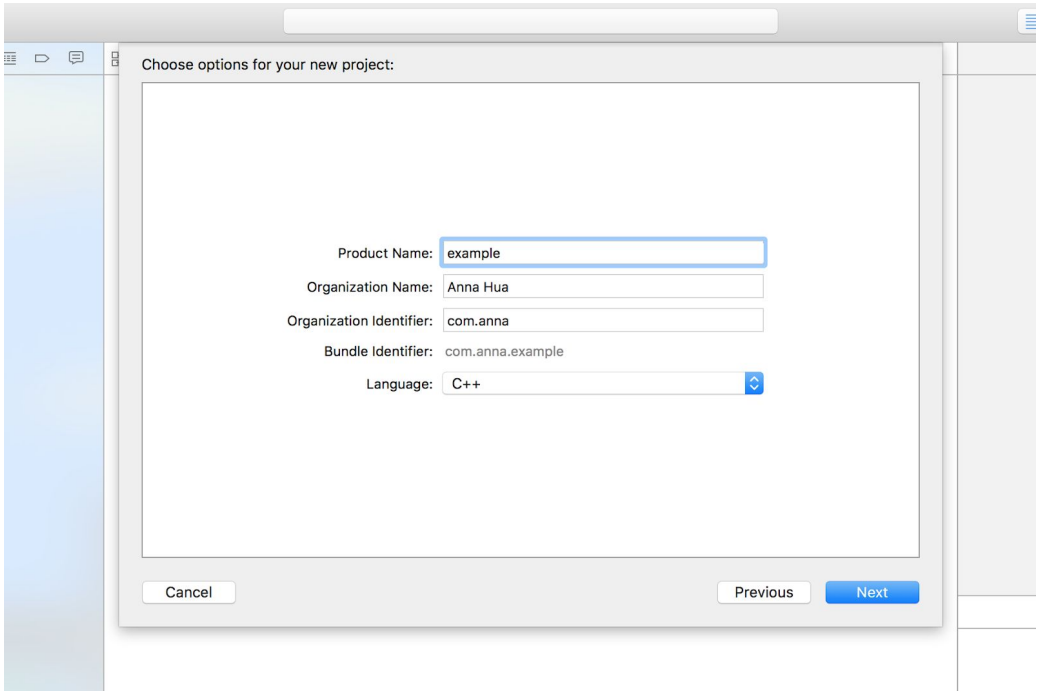
Creating Projects

To create a project, open Xcode, and go to File->New->Project.



In the OS X section, select "Application" then select the "Command Line Tool" option. Press "Next". Put the project name under "Product Name" and change "Language" to C++. Leave the other fields the way they are, they're not important for this course.

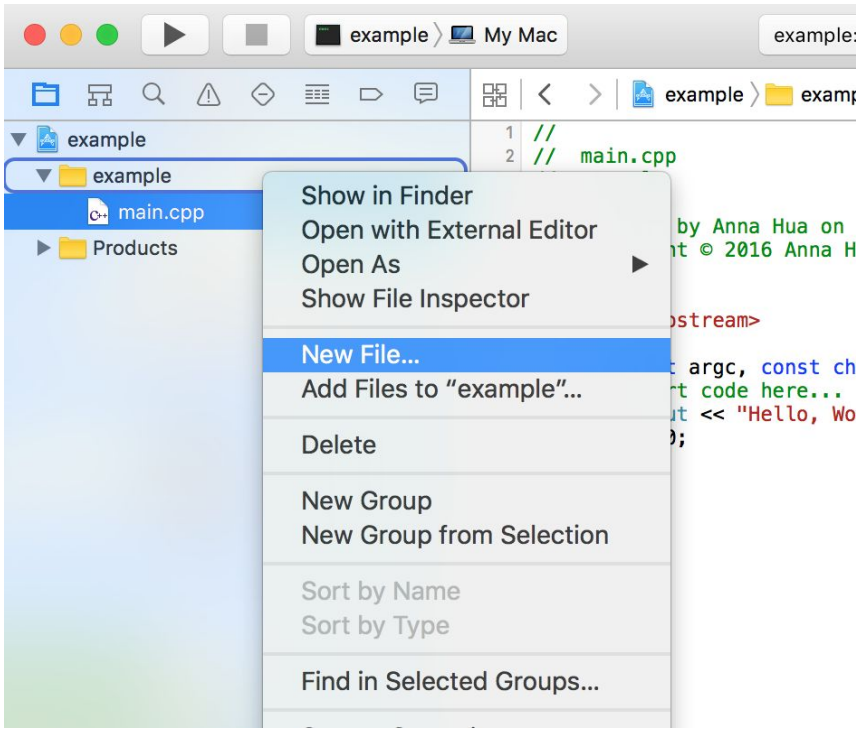




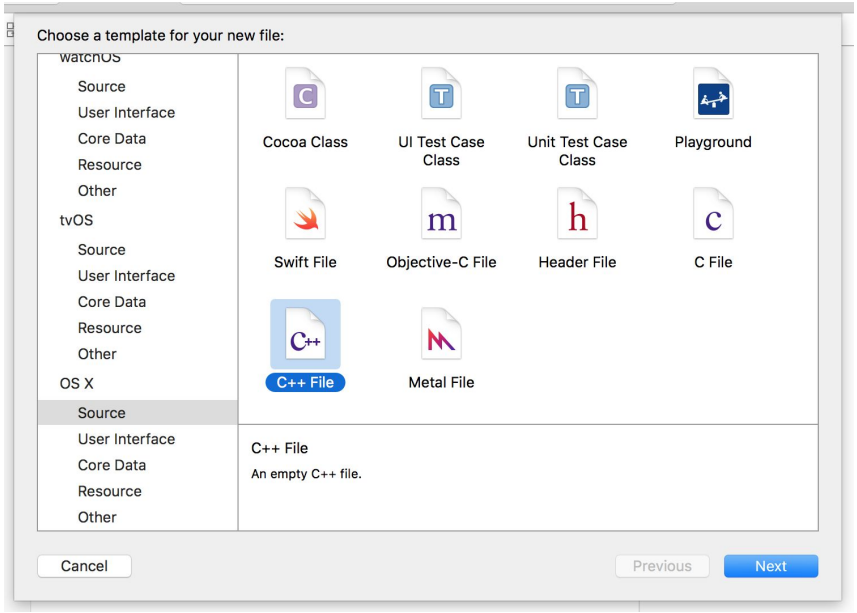
On the next screen, select where you want to save your project. If you want to use Xcode for source control, check the “Create Git repository on…” box. You can also use the command line for git version control.

Adding Source Code

To add a new file, control-click (right click) anywhere on the right panel and select “New File”.



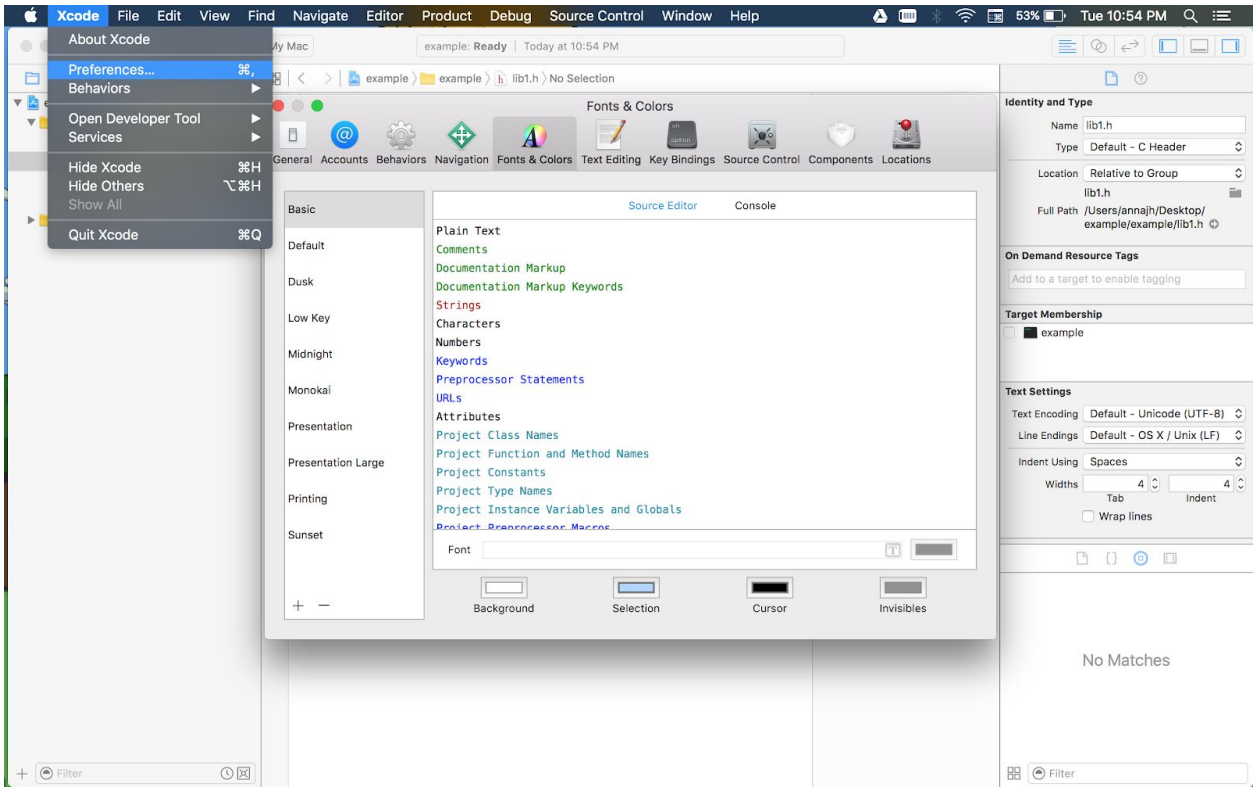
To add a header and/or .cpp file, go to the “OS X” section and select “Source”. There you can select “Header File” or “C++ File”.



You can also add existing files to the project by selecting “Add Files to <project name>” instead of “New Files”. A convenient place to store them is in the same directory as other source code, which you can find by selecting “Show in Finder” (look at a couple pictures before this). Note that the file structure that you see in the Xcode IDE is not necessarily the actual file structure on your machine.

Make it Pretty

You can change the color scheme, font size, add line numbers, etc. by going to Xcode->Preferences…



Showing line numbers and other very useful features are found under the “Text Editing” section.

Building

Source Code Files

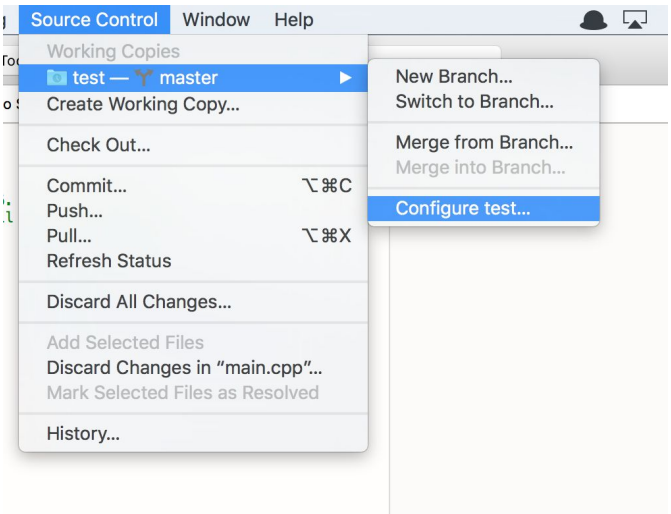
If you need to upload your files, you can look in the project folder on your computer. You specified the project location when you created the project in Xcode - this is where your source code files live.

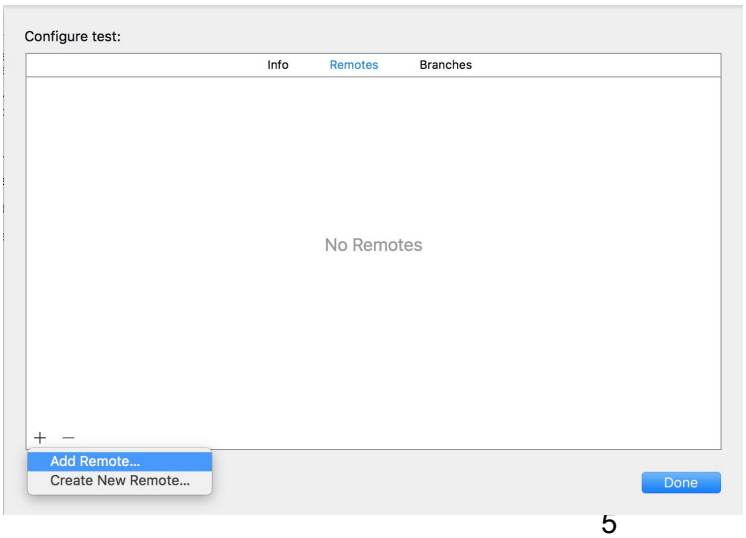
Rather than having multiple copies of your files, on your Mac, on CAEN for compiling there, on a flash drive, etc., it would be better to keep them in one place. Consider using the EECS gitlab server: https://gitlab.eecs.umich.edu. Xcode is compatible with it! If you look back a few pages to when we created the project, there is an option to create a local Git repository for your project (this option is selected by default when you create a project). We want to publish your local repository to the remote gitlab server.

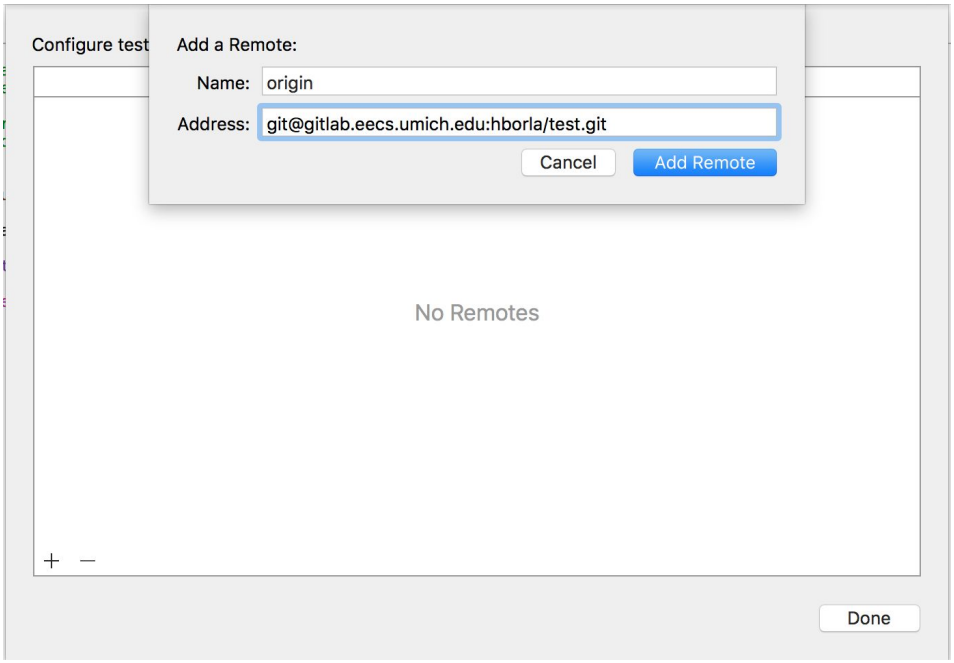
Before you can upload your code to the GitLab server, you must first establish a secure connection between your computer and GitLab. Steps to do this can be found here: http://docs.gitlab.com/ce/ssh/README.html

From here, create a new repository for your project on GitLab. You may want to name the repository the same as the project name. This is where all of your code will be stored.

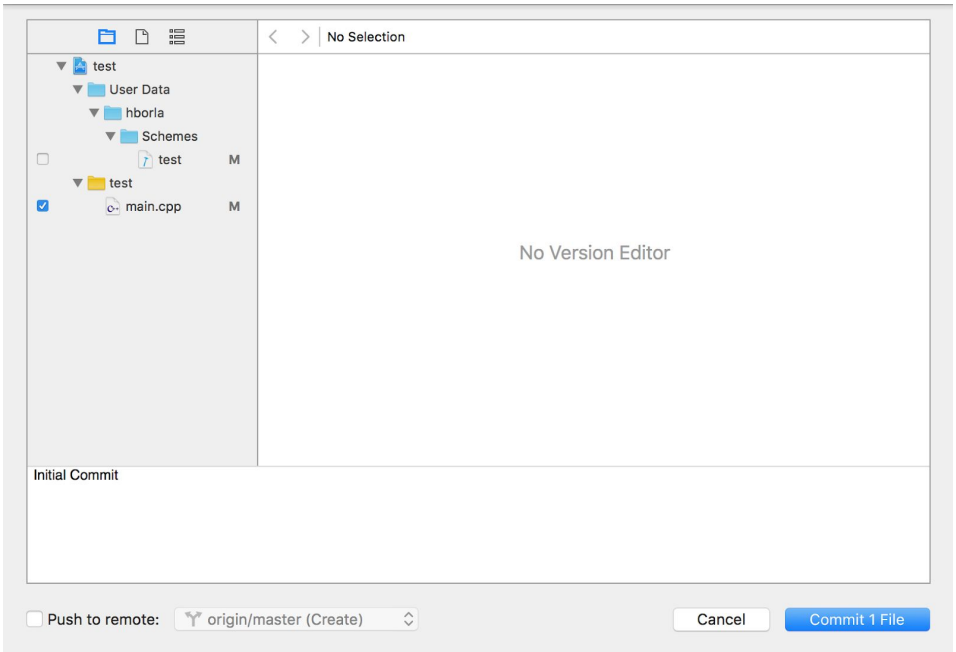
Before you can upload your code to the server, you must first configure the Source Control settings in Xcode. Go to Source Control -> <project name - branch name> -> Configure <project name>, click on the “Remotes” tab and hit the plus sign to add a remote location. Leave the default name as “origin” and under the “Address” section, type in “git@gitlab.eecs.umich.edu:<uniqname>/<repository name>.git” then click “Add Remote”. In the example shown below, the uniqname is “hborla” and the repo/project name is “test”







You can now upload your code to the server. In Xcode, under the “Source Control” menu bar tab, click “Commit…”; you can then select the files you want to upload and add a commit message. When you’re done, click “Commit <N> files”.



Finally, under Source Control -> Push … click “Push” to upload your selected files to the server. Change the “Username and Login” to the “SSH Keys” option and type in the password that you used when generating your SSH key. When you log into CAEN, you can clone your GitLab repository and simply do a “git pull” each time you update your code in Xcode.

Command-Line Parameters

After creating a project, you may want to pass command-line parameters when you run the program, such as flags, filenames, etc. To do this, type Command + < (or go to Product -> Scheme -> Edit Scheme), click on the Arguments tab, and add your command line arguments under the “Arguments Passed On Launch” section.

I/O Redirection

For Project 1 at the command line, you could type in:

```
-q < test-1-qm.txt
```

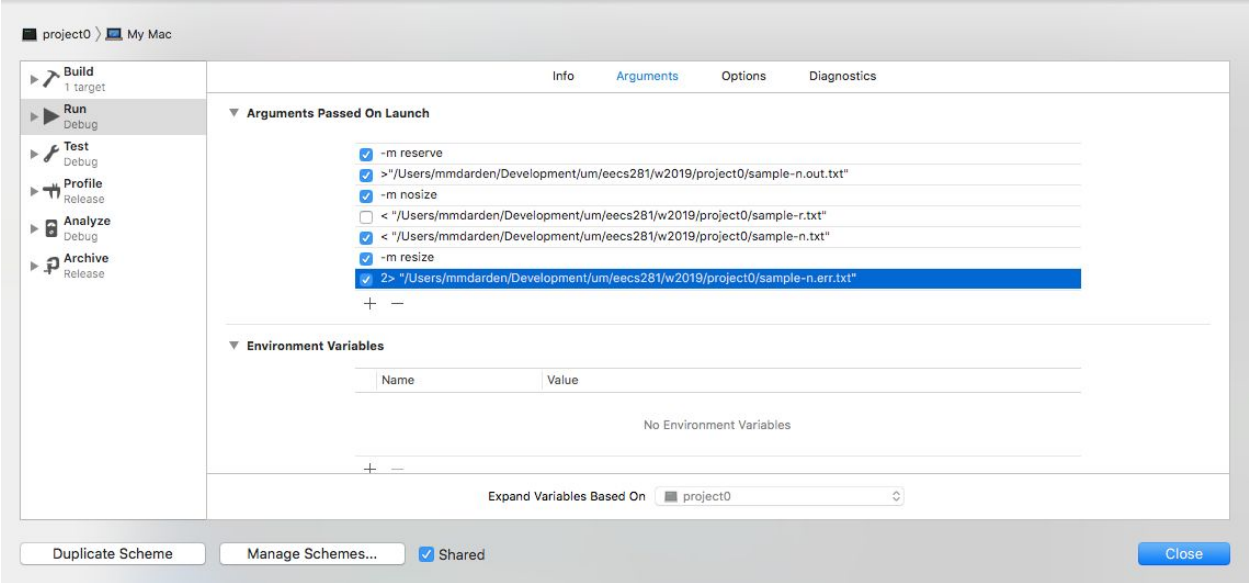
Where test-1-qm.txt is the name of a text file inside of your project folder, that will be used as input. To create the input file, create a new file as you normally would in Xcode, but when it prompts you for the file type, click “Other” on the left side. Put the file in the same folder as your source code files, name your file with the appropriate extension, and then you can edit the file right in Xcode.

Unfortunately, Xcode has no way to automatically redirect input or output. We have to add code to do it. We’ve provided a file “xcode_redirect.hpp” that you should add to your EECS 281 projects to facilitate redirection and IDE debugging.

Once that’s done, include the following line of code at the top of your main().

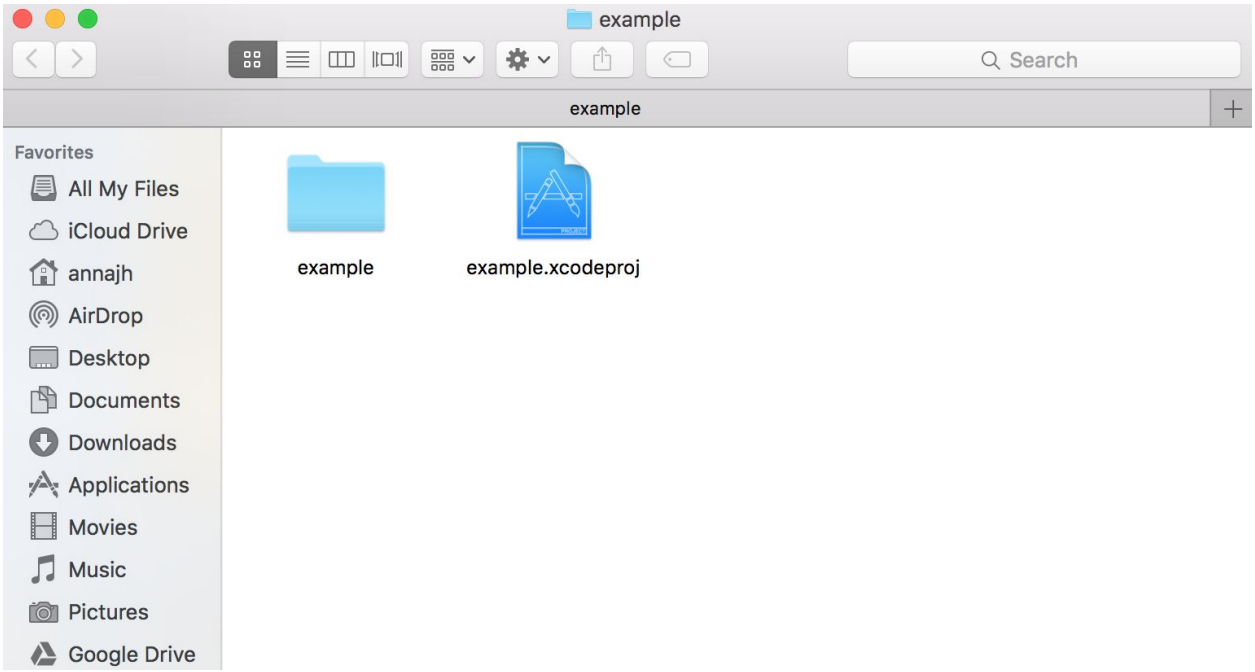
```
xcode_redirect(argc, argv);
```

You can then edit your project scheme to include redirected input, output, and error messages. Be sure to use absolute pathnames, to make sure that file access works!



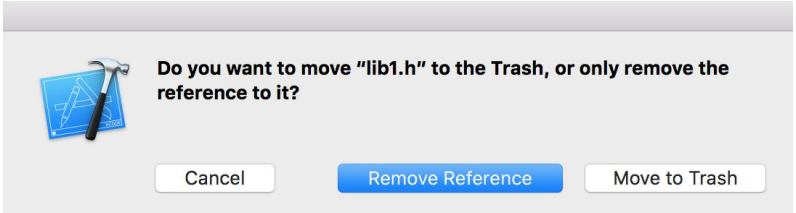
Opening a Project

If the project that you want to open has scrolled off your list of recently opened projects, go to File->Open… or, alternatively, you can go into your project directory and open the directory with .xcodeproj extension. In the picture below, you would open example.xcodeproj.



Other Tips

If you delete a file in Xcode this confirmation will pop up

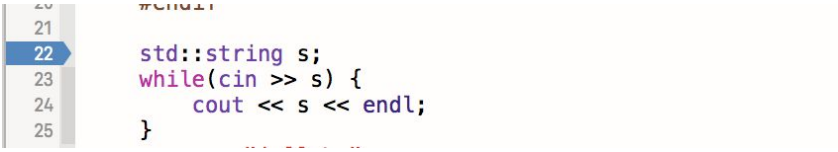


If you're concerned that you may want to add the file back into the project, consider selecting "Remove Reference" to eliminate the hassle of searching through Trash.

On the upper-right corner are several options for viewing

Play around with them a little bit, some of them are very useful for coding and developing.

You can set a breakpoint in your code just by clicking the line number. A line number highlighted in blue indicates a breakpoint:



When you run your code, you can use the visual debugger to step through your code and view the state of the program from the breakpoint. You can click the arrow next to the variable names to view more information.