OpenGL Environment Setup

Computer Graphics

CGV Lab, NTHUCS

Prerequisite

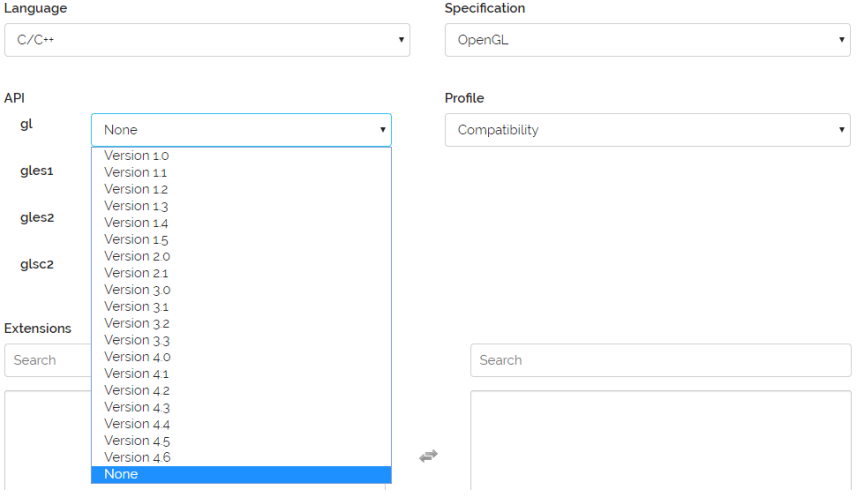
- **On Windows:**
  - **Microsoft Visual Studio 2017 community/professional**
- **On Mac:**
  - **Xcode 11**
- **Download GLFW & GLAD from official website.**
- **https://www.glfw.org/**
- **https://glad.dav1d.de/**
- **Download the sample code on iLMS.**

GLFW

- GLFW is an Open Source, multi-platform library for OpenGL, OpenGL ES and Vulkan development on the desktop.
- It provides a simple API for creating windows, contexts and surfaces, receiving input and events.

GLAD

- GLAD is an OpenGL loading library that loads pointers to OpenGL functions at runtime, core as well as extensions.
- It generates a loader for your exact needs based on the official specifications from the Khronos SVN. This means they are always up to date!

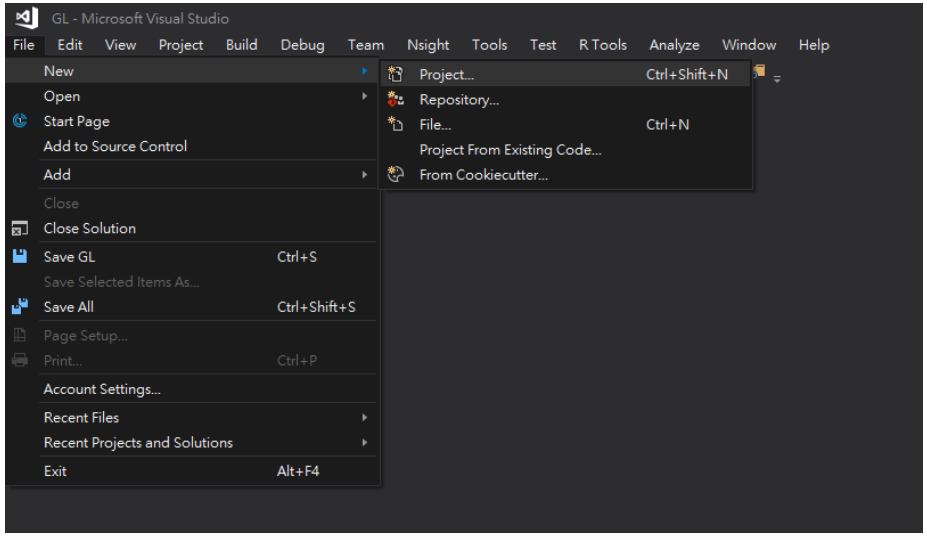


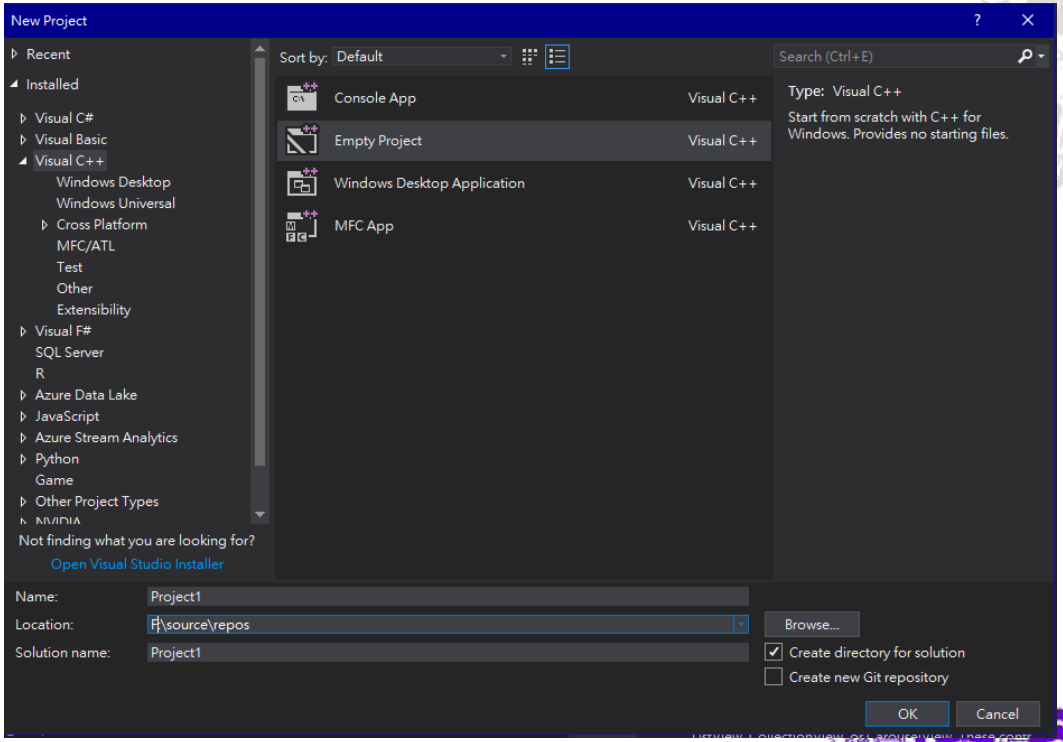
Visual Studio

- **Ensure you have installed C++ package when installing IDE.**

Visual Studio

- Create a new empty project.





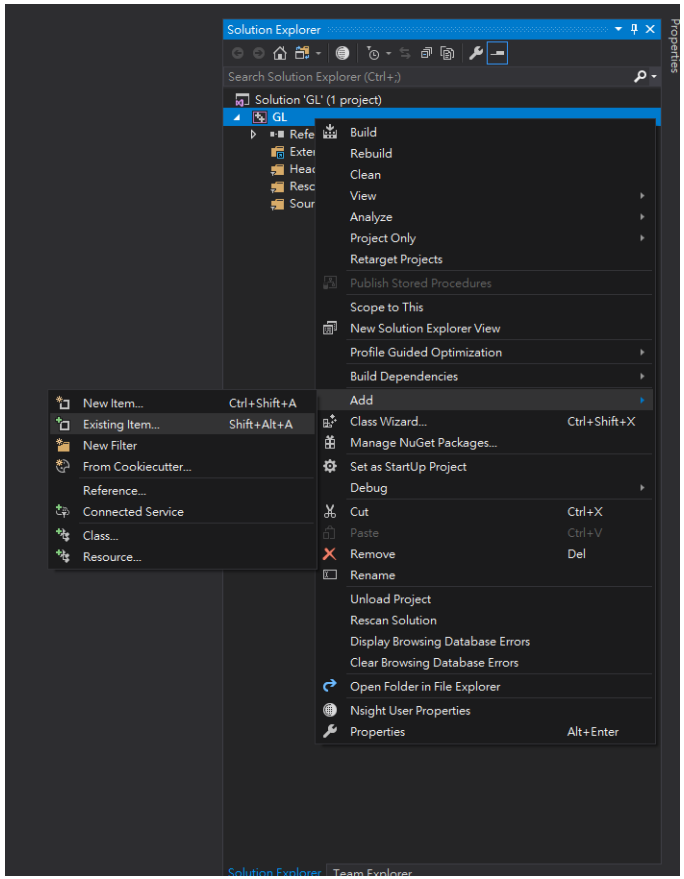
Visual Studio

- **Copy the sample code to project folder.**

Recommand project hierarchy
-projectName.sln
-projectName
- **projectName.vcxproj**
- **projectName.vcxproj.filters**
- **projectName.vcxproj.user**
- **main.cpp**
- **shader.fs**
- **shader.vs**
- **textfile.cpp**
- **textfile.h**

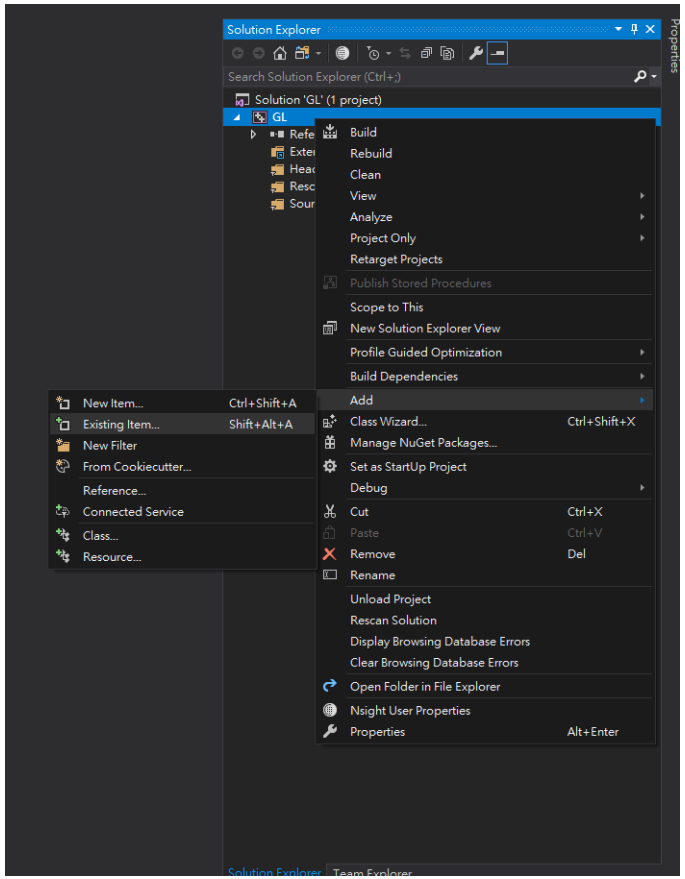
Visual Studio

- **Add sample code to project.**



Visual Studio

- **Add sample code to project.**



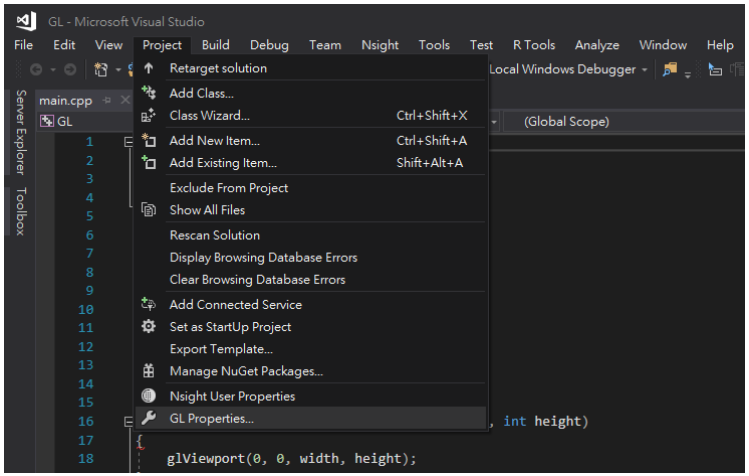
Visual Studio

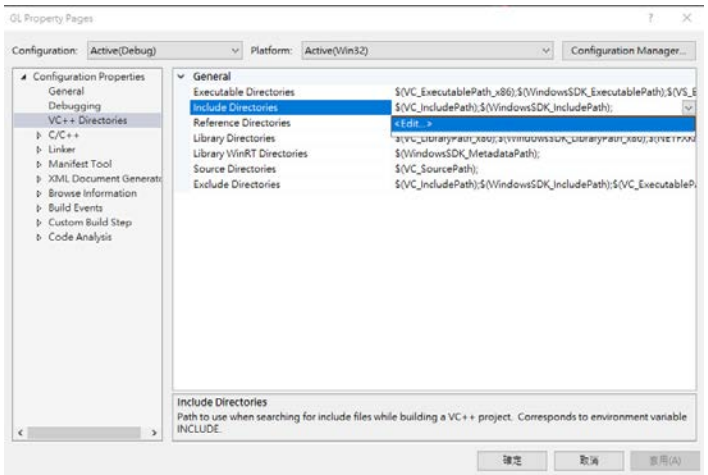
- **Copy the GLFW & GLAD to project folder.**
- **glad.c can be directly added to project and compiled.**
- **You can compile glfw3.lib from source or download pre-compiled version from official website.**

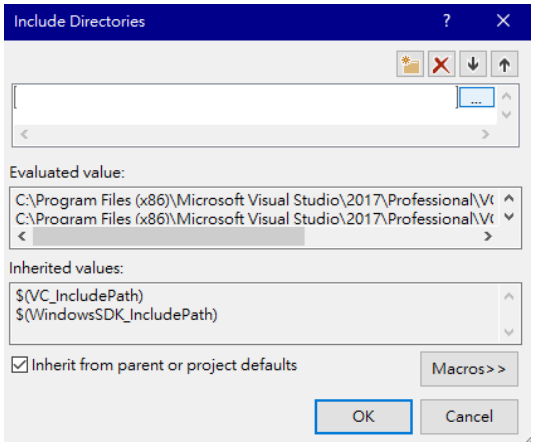
Recommand project hierarchy
-projectName.sln
-include
- glad
- GLFW
- KHR
-projectName
- glad.c
-lib
- glfw3.lib

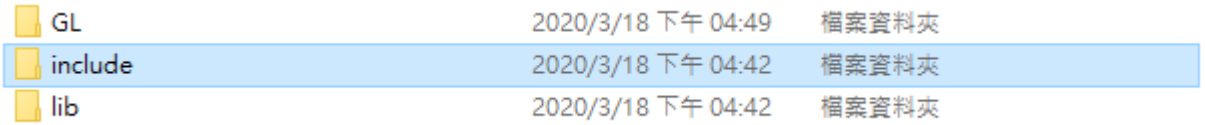
Visual Studio

- Add include path to project setting.



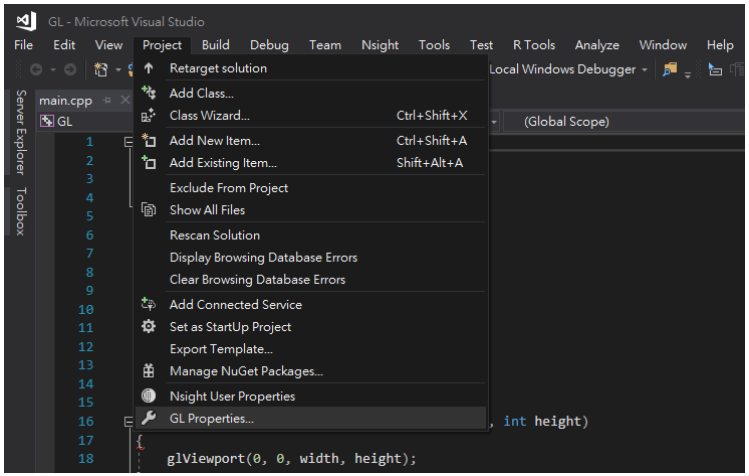


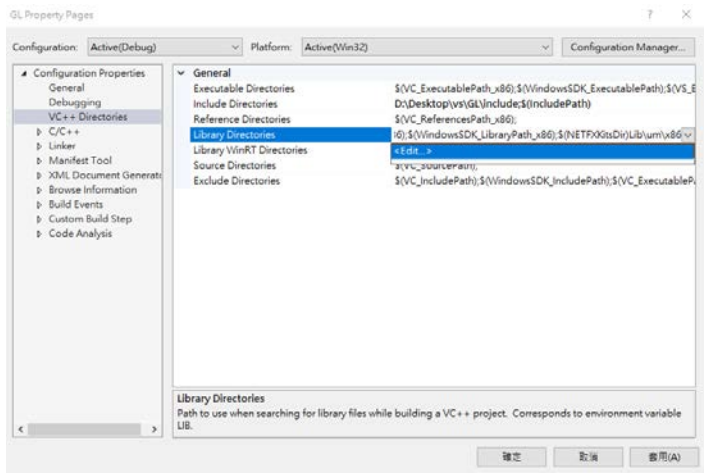


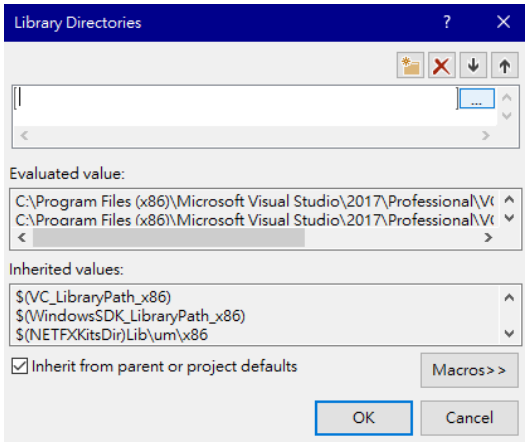


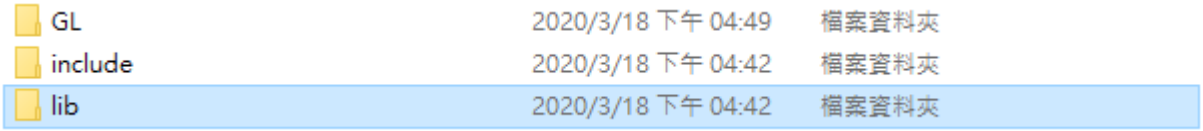
Visual Studio

- Add library path to project setting.



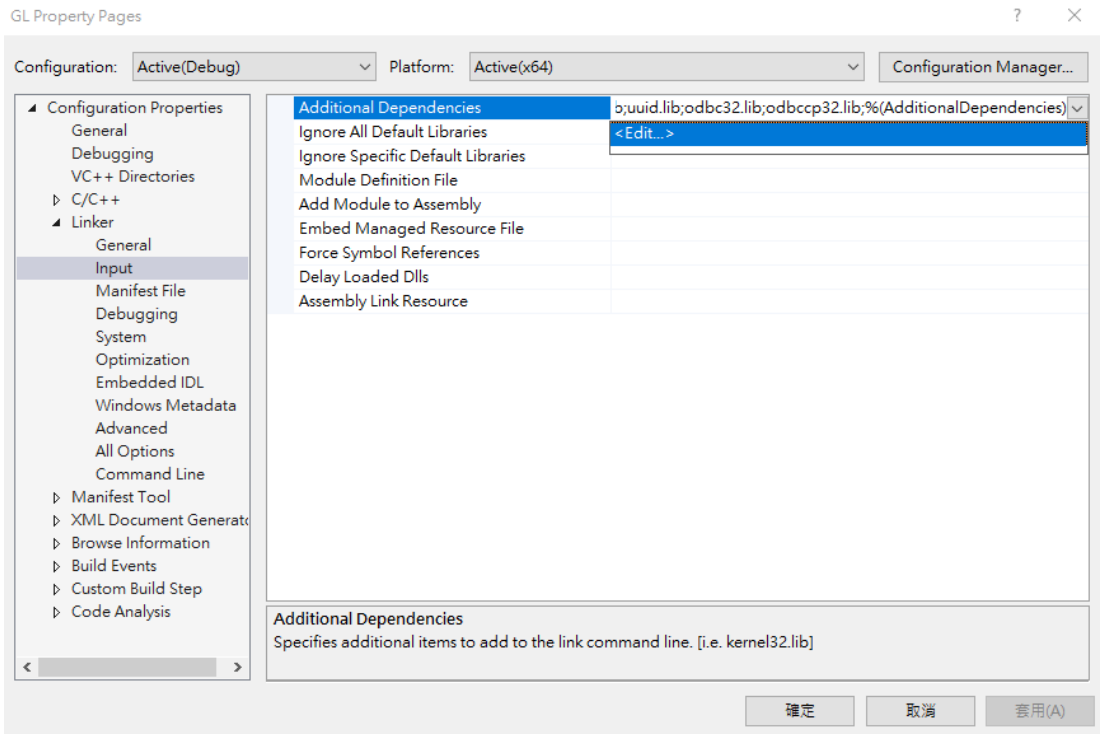


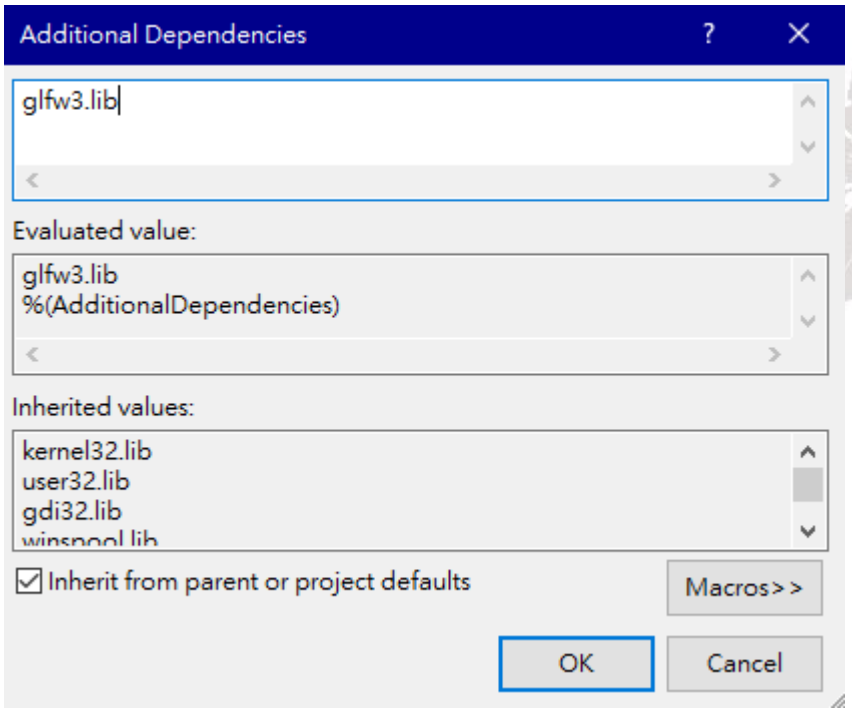




Visual Studio

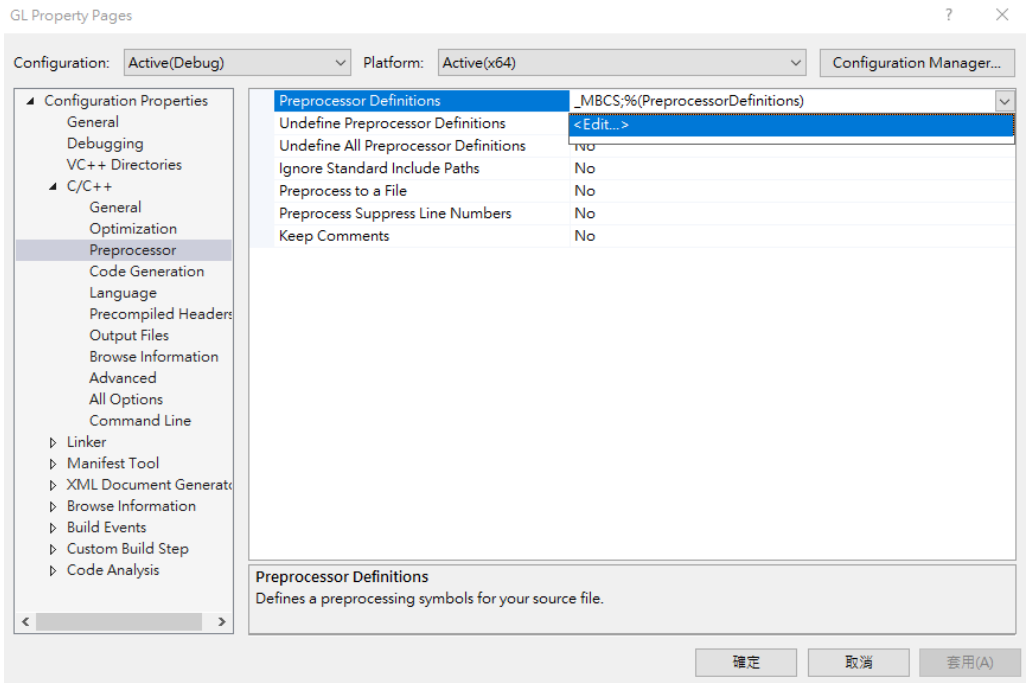
- **Add glfw3.lib to linker input.**





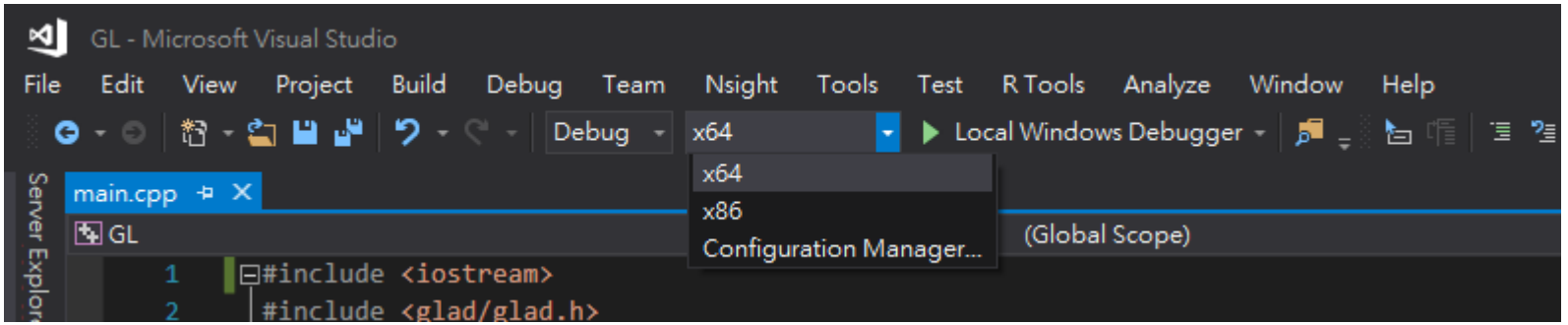
Visual Studio

- Add _CRT_SECURE_NO_WARNINGS to preprocessor definitions to prevent secure warning.



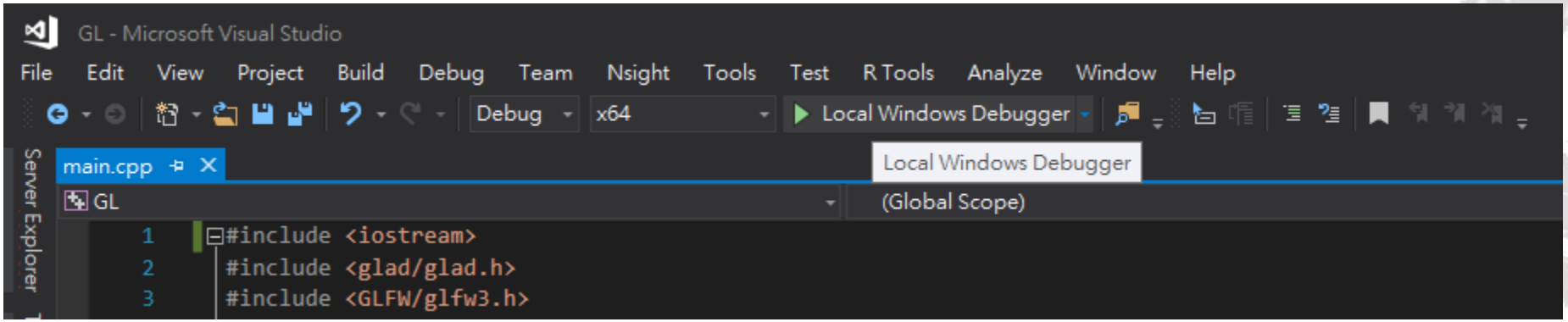
Visual Studio

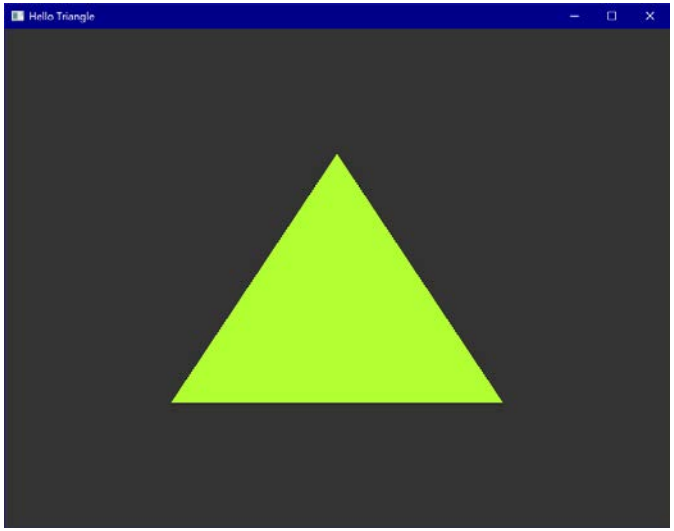
- Choose a proper configuration.
- If you use 32-bit GLFW, choose x86.
- If you use 64-bit GLFW, choose x64.



Visual Studio

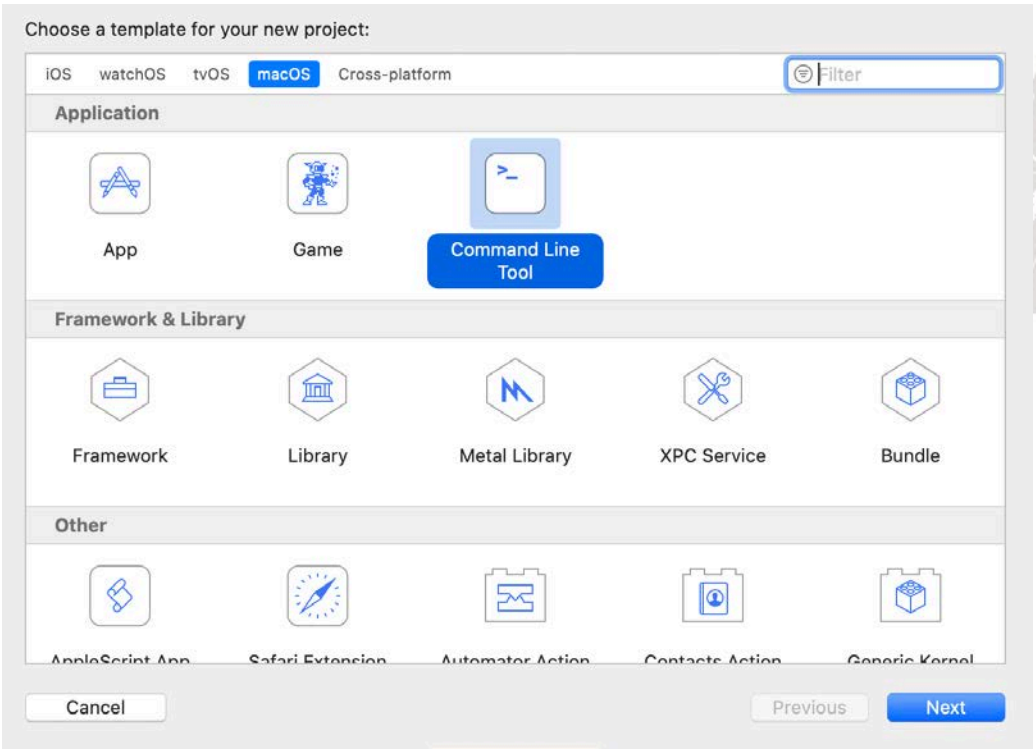
- **Build and Run.**





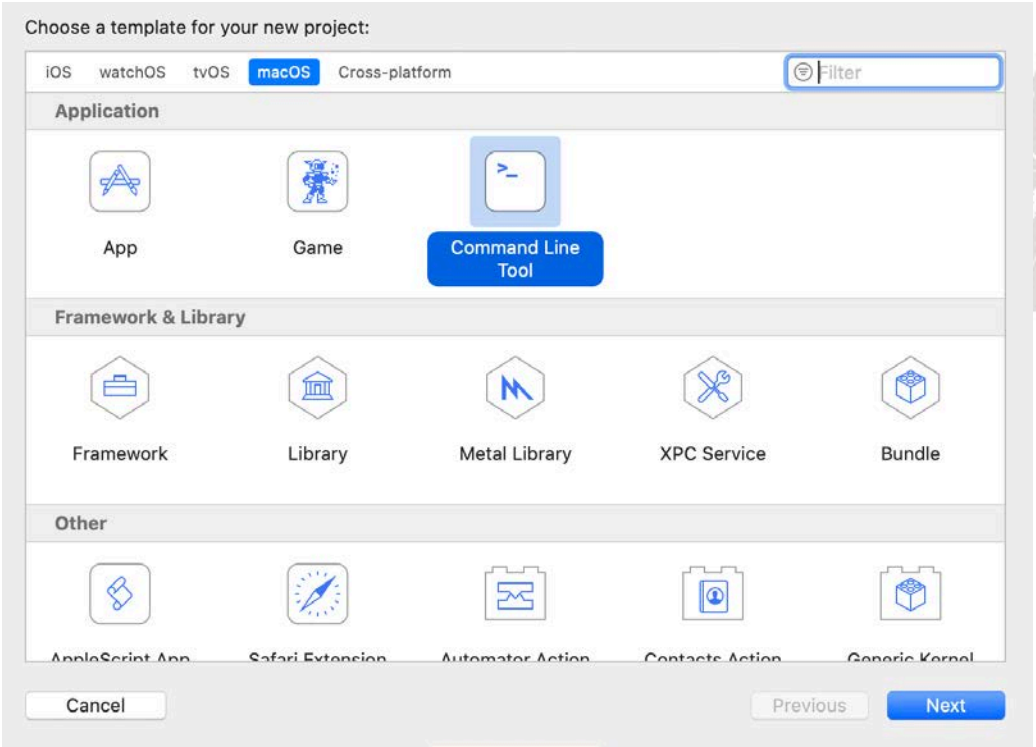
Xcode

- **Create new Xcode command line project.**



Xcode

- **Create new Xcode command line project.**



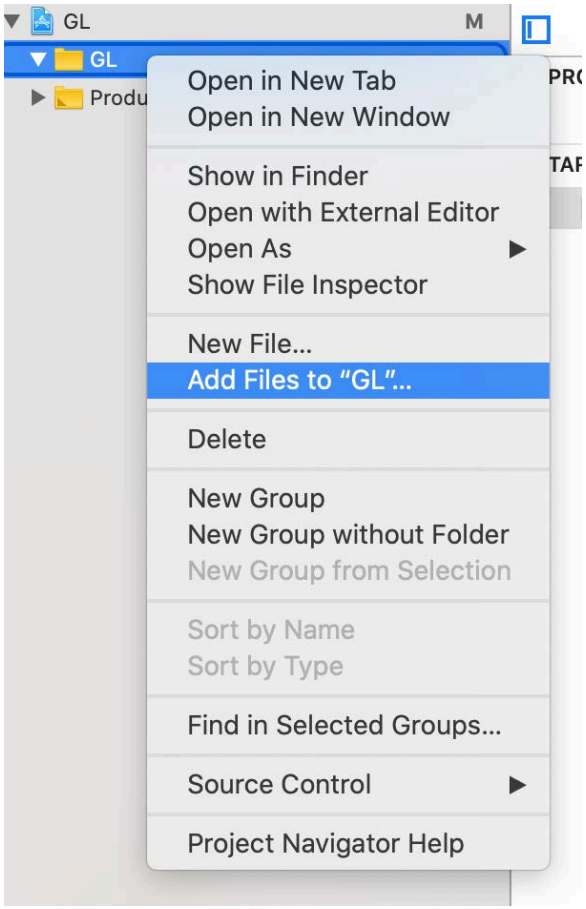
Xcode

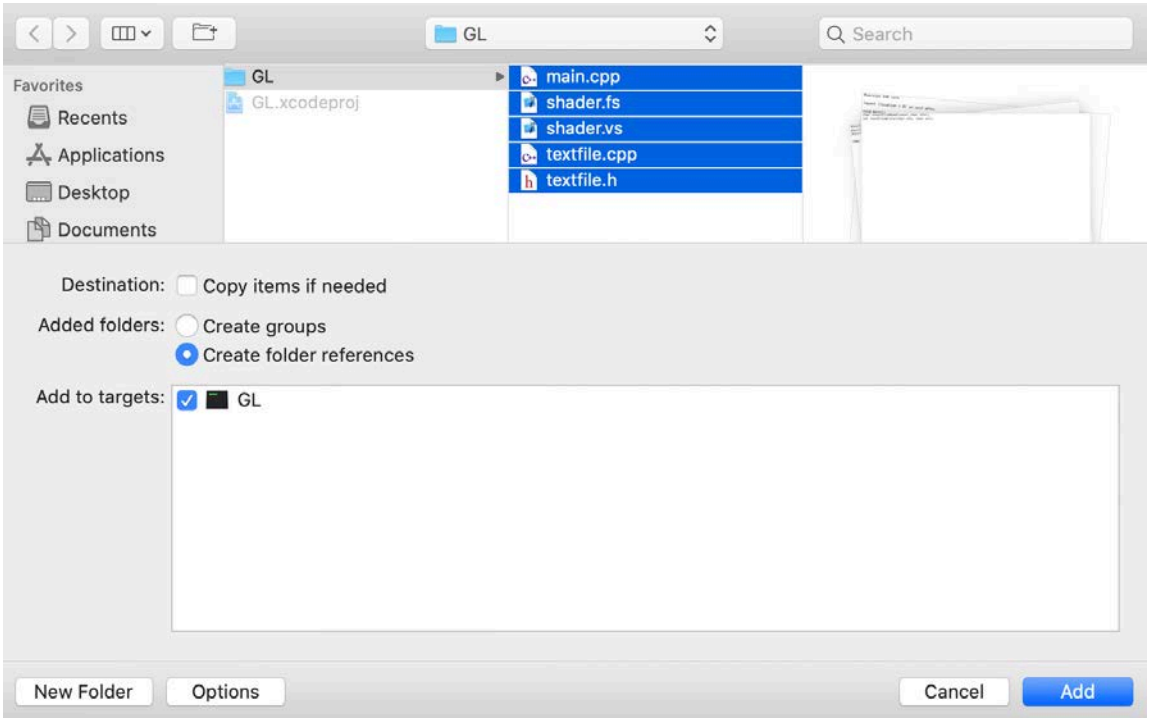
- **Copy the sample code to project folder.**

Recommand project hierarchy
-projectName.xcodeproj
-projectName
- **main.cpp**
- **shader.fs**
- **shader.vs**
- **textfile.cpp**
- **textfile.h**

Xcode

- Add sample code to project.





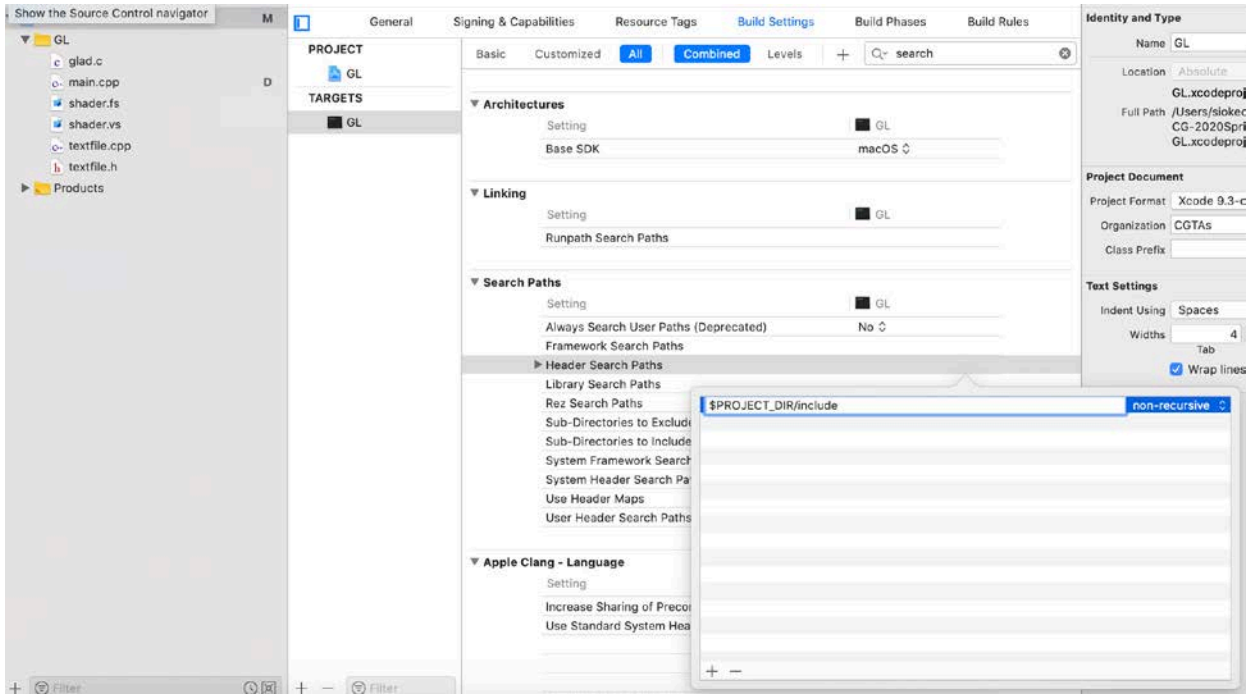
Xcode

- Copy the GLFW & GLAD to project folder.
- glad.c can be directly added to project and compiled.
- You can compile glfw3.dylib from source or download pre-compiled version from official website.

Recommand project hierarchy
-projectName.xcodeproj
-include\
    - glad\
    - GLFW\
    - KHR\
-projectName\
    - glad.c
-lib\
    - glfw3.dylib

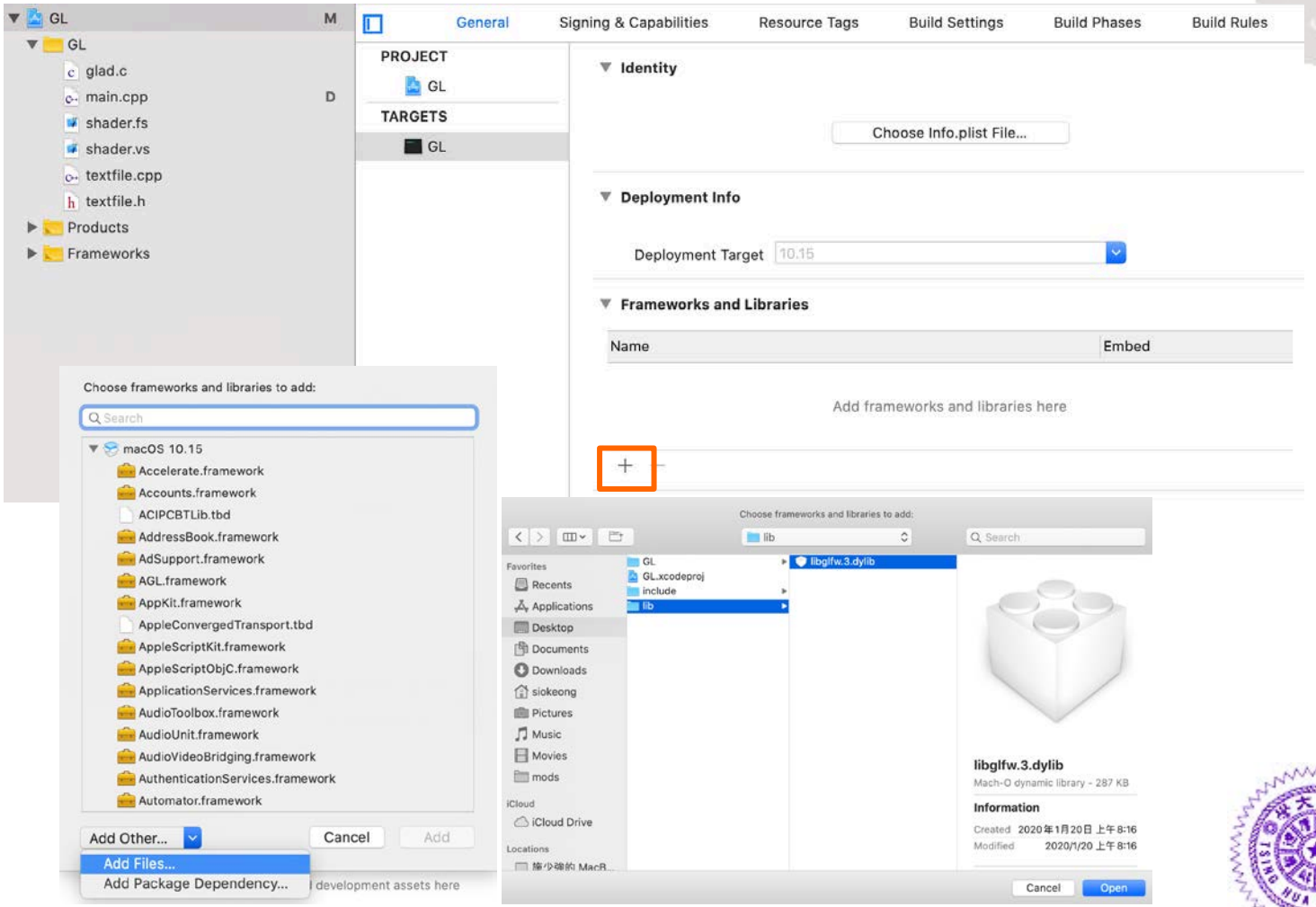
Xcode

- Add include path to header search paths.
- $PROJECT_DIR means the location of projectName.xcodeproj.



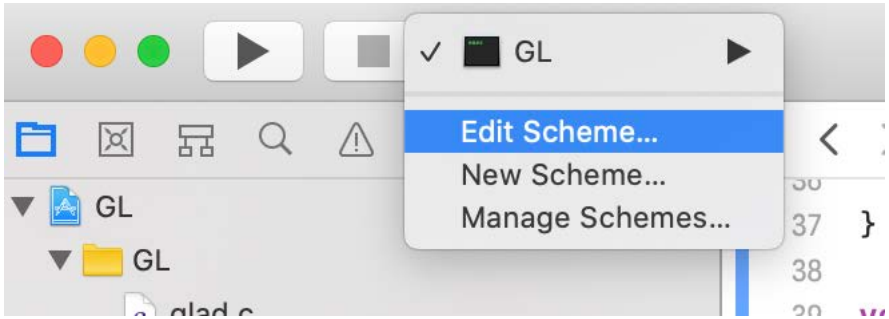
Xcode

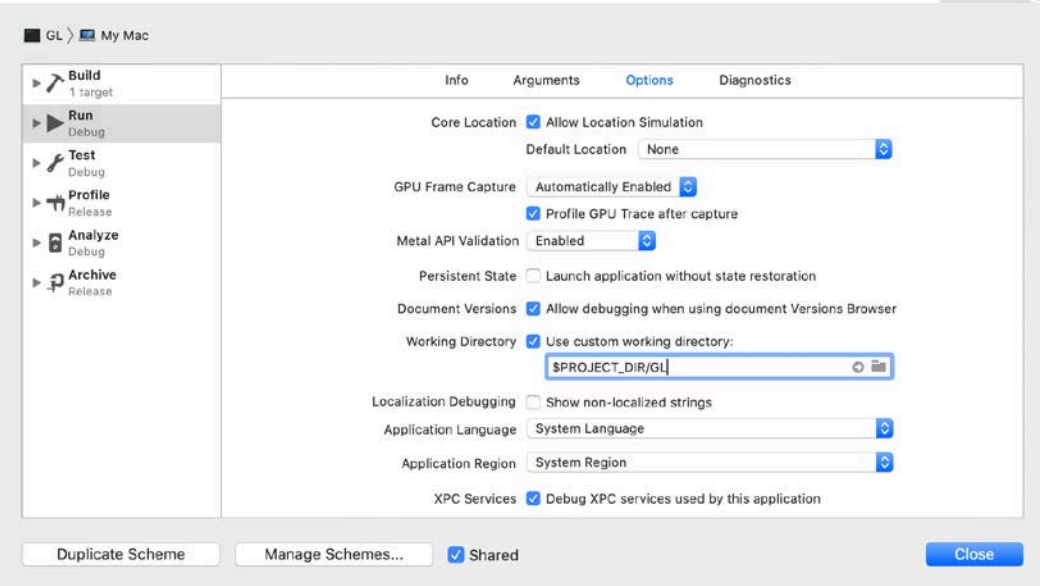
- **Add lib file to libraries list.**



Xcode

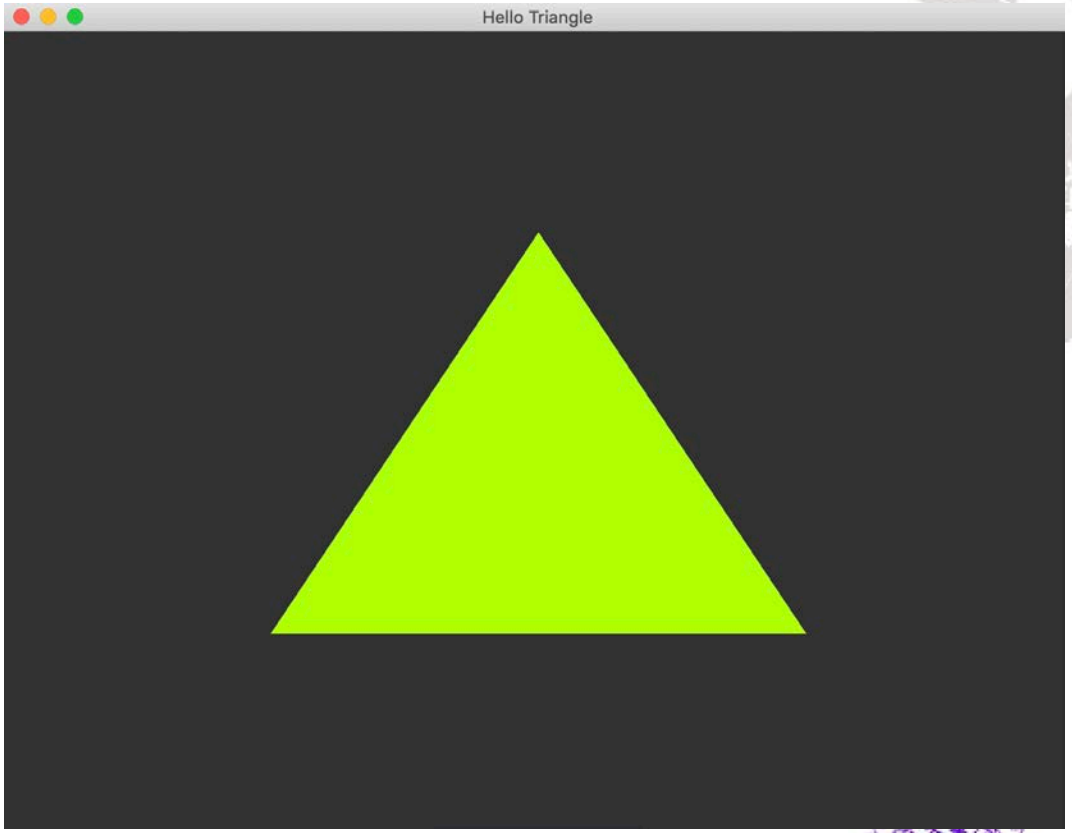
- Set the working directory to correct folder.
- We recommend set the path to $PROJECT_DIR/projectName.





Xcode

- **Build and run.**



Sample Code

- **Download “HelloTriangle.zip” available from iLMS, unzip the downloaded file.**
- **The file contains a framework already setup for VS2017 and Xcode, choose a framework depends on your OS.**

Reference

- https://www.glfw.org/
- https://www.khronos.org/opengl/wiki/OpenGL_Loading_Library
- https://glad.dav1d.de/