



Implementation: Start Up

Introduction to Computer Graphics

Yu-Ting Wu

1

Library

2

Library

- **GLUT: OpenGL Utility Toolkit** ([link](#))
 - Window system independent
 - Implement a simple window application programming interface (API) for OpenGL
 - Designed for constructing small to medium-sized OpenGL programs
 - For large applications, it is suggested to use a native window system toolkit such as Qt for more sophisticated UI
- **FreeGLUT: Free OpenGL Utility Toolkit** ([link](#))
 - GLUT has gone into stagnation and has some issues with licenses
 - FreeGLUT is intended to be a full replacement for GLUT

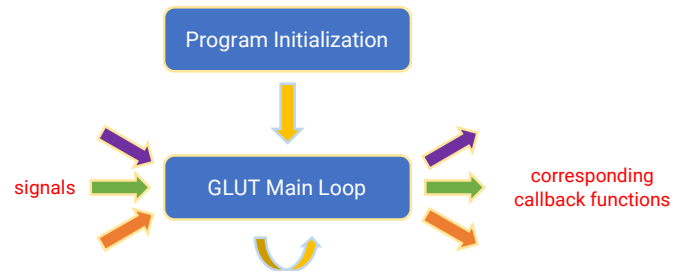
3

Program

4

Program Structure Overview

- OpenGL programs are event-driven



5

The First Program

```
// OpenGL and FreeGlut headers.
#include <freeglut.h>
int main(int argc, char** argv)
{
    // Setting window properties.
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA | GLUT_DEPTH);
    glutInitWindowSize(640, 360);
    glutInitWindowPosition(100, 100);
    glutCreateWindow("OpenGL Renderer");

    // Initialization.
    SetupRenderState();

    // Register callback functions.
    glutDisplayFunc(RenderSceneCB);
    glutIdleFunc(RenderSceneCB);
    glutReshapeFunc(ReshapeCB);
    glutSpecialFunc(ProcessSpecialKeysCB);
    glutKeyboardFunc(ProcessKeysCB);

    // Start rendering loop.
    glutMainLoop();

    return 0;
}
```

create the window
and set window
properties

do initialization
jobs

register callback
functions

start the
main loop

6

Create a OpenGL (GLUT) Window

- `void glutInit(int *argc, char **argv);`
 - Initialize the GLUT library

```
glutInit(&argc, argv);
```
- `int glutCreateWindow(char *name);`
 - Create a top-level window

```
glutCreateWindow("OpenGL Renderer");
```

7

Setting Window Properties

- `void glutInitWindowSize(int width, int height);`
 - Set the initial window size
- `void glutInitWindowPosition(int x, int y);`
 - Set the initial window position

```
glutInitWindowSize(640, 360);
glutInitWindowPosition(100, 100);
```
- `void glutInitDisplayMode(unsigned int mode);`
 - Set the initial display mode
 - <https://www.opengl.org/resources/libraries/glut/spec3/node12.html>

```
glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA | GLUT_DEPTH);
```

8

Setting Callback Functions

- Register the callback functions when receiving events
- Commonly used
 - `glutDisplayFunc`
 - `glutIdleFunc`
 - `glutReshapeFunc`
 - `glutKeyboardFunc` / `glutSpecialFunc`
 - `glutMouseFunc`
 - `glutMenuStatusFunc`
- Each callback function has its own input format
- Please refer to the following page for all possible callback functions
 - <https://www.opengl.org/resources/libraries/glut/spec3/node45.html>

9

Setting Callback Functions (cont.)

```
void RenderSceneCB()
{
    glClearColor(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    // Render something here.
    // TODO.
    glutSwapBuffers();
}

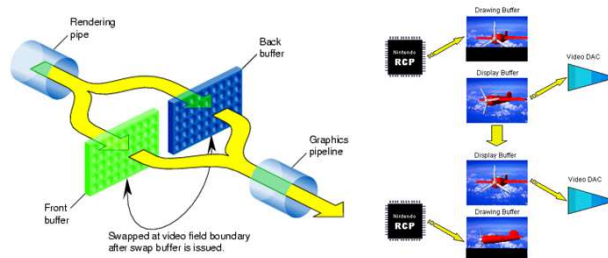
void ProcessKeysCB(unsigned char key, int x, int y)
{
    // Handle other keyboard inputs those are not defined as special keys.
    if (key == 27) { ESC
        // Release memory allocation if needed.
        exit(0);
    }
}
```

10

10

Double Buffers

- Prevent artifacts due to potentially seeing parts of an incomplete frame (that is currently drawn)
 - Set the display mode to **GLUT_DOUBLE** in the `glutInitDisplayMode` function
 - Call `glutSwapBuffers` after rendering finished



11

11

Initialization

- `void glClearColor(GLfloat red, GLfloat green, GLfloat blue, GLfloat alpha);`
- Set the color to clear the color buffer

```
void SetupRenderState()
{
    float clearColor[4] = {0.44f, 0.57f, 0.75f, 1.00f};
    glClearColor(
        (GLfloat)(clearColor[0]),
        (GLfloat)(clearColor[1]),
        (GLfloat)(clearColor[2]),
        (GLfloat)(clearColor[3])
    );
}
```

12

12

Start the Main Rendering Loop

- `void glutMainLoop(void);`
 - Enter the GLUT event processing loop
 - OpenGL programs are event-driven



13

13

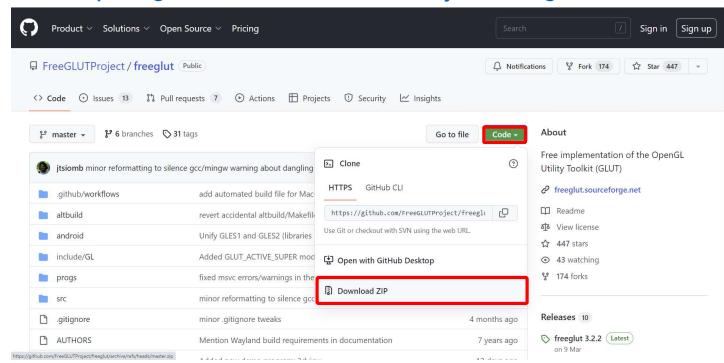
Build Binaries of FreeGLUT with Visual Studio

14

14

FreeGLUT

- Download the source code from <https://github.com/FreeGLUTProject/freeglut>

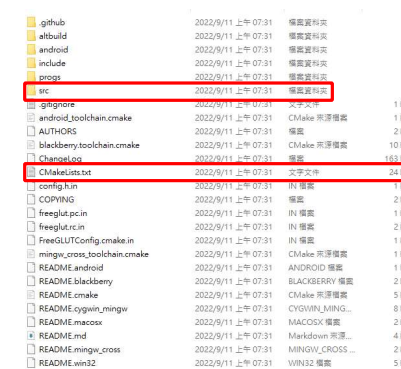


15

15

FreeGLUT (cont.)

- Unzip the package



Build the source code using CMake

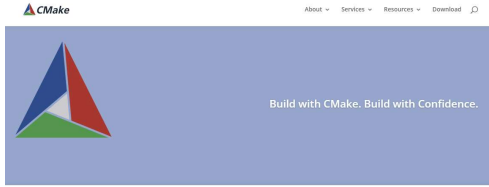
16

16

Introduction to Computer Graphics 2022

CMake

- Download and install CMake: <https://cmake.org/>



Build with CMake. Build with Confidence.

CMake is an open-source, cross-platform family of tools designed to build, test and package software. CMake is used to control the software compilation process using simple platform and compiler independent configuration files, and generate native makefiles and workspaces that can be used in the compile environment of your choice. The suite of CMake tools were created by Kitware in response to the need for a powerful, cross-platform build environment for open-source projects such as ITK and VTK.

CMake is part of Kitware's collection of commercially supported **open-source platforms** for software development.

Download Latest Release
Visit the download page

Support and Services
Get support or consulting service for CMake

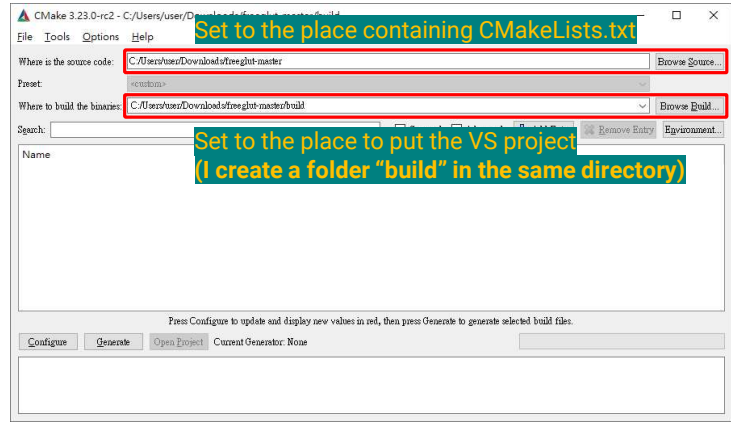
Contact Us
Have a question about a CMake project? We can help

17

17

Introduction to Computer Graphics 2022

Setup CMake for Building FreeGLUT



Set to the place containing CMakeLists.txt

Set to the place to put the VS project
(I create a folder "build" in the same directory)

Press Configure to update and display new values in red, then press Generate to generate selected build files.

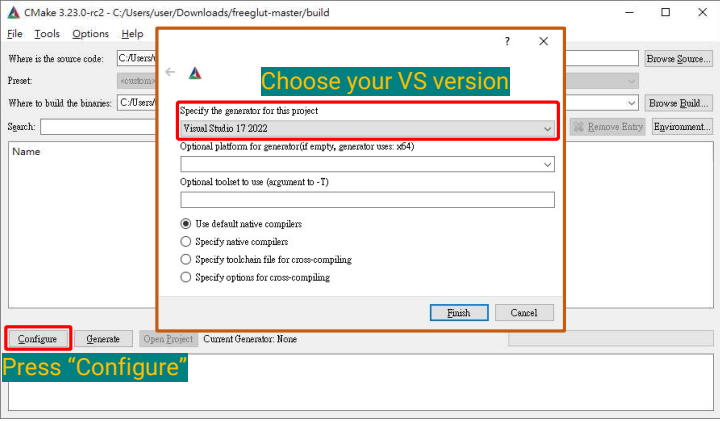
Configure Generate Open Project Current Generator: None

18

18

Introduction to Computer Graphics 2022

Configuration



Choose your VS version

Specify the generator for this project

Visual Studio 17 2022

Optional platform for generators (if empty, generator uses: x64)

Optional toolset to use (argument to -T)

☒ Use default native compilers
☐ Specify native compilers
☐ Specify toolchain file for cross-compiling
☐ Specify options for cross-compiling

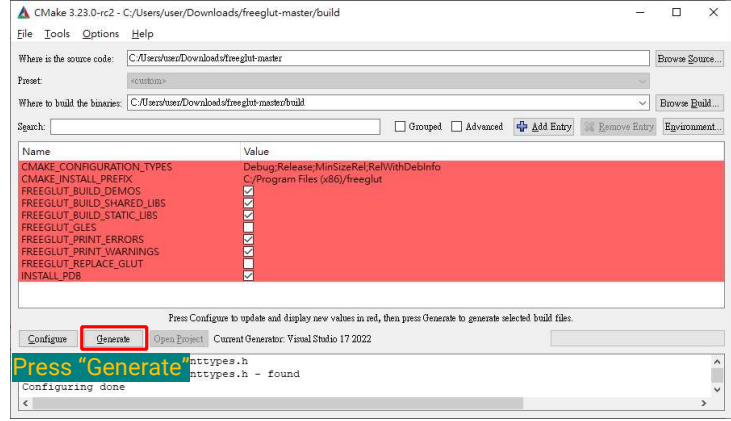
Press "Configure"

19

19

Introduction to Computer Graphics 2022

Generate VS Project



Press "Generate"

Configuring done

20

20

Examine VS Project

名稱	修改日期	類型	大小
Fractals_random.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB
Fractals_random_static.vcxproj	2022/9/14 下午 03:47	VC++ Project	60 KB
Fractals_random_static.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB
Fractals_static.vcxproj	2022/9/14 下午 03:47	VC++ Project	60 KB
Fractals_static.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB
freeglut.pc	2022/9/14 下午 03:46	PC 檔案	1 KB
freeglut.rc	2022/9/14 下午 03:46	Resource Script	2 KB
freeglut.sln	2022/9/14 下午 03:47	Visual Studio Sol...	43 KB
freeglut.vcxproj	2022/9/14 下午 03:47	VC++ Project	64 KB
freeglut.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	8 KB
freeglut_static.vcxproj	2022/9/14 下午 03:47	VC++ Project	59 KB
freeglut_static.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	8 KB
indexed_color.vcxproj	2022/9/14 下午 03:47	VC++ Project	59 KB
indexed_color.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB
indexed_color_static.vcxproj	2022/9/14 下午 03:47	VC++ Project	60 KB
indexed_color_static.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB
INSTALL.vcxproj	2022/9/14 下午 03:47	VC++ Project	10 KB
INSTALL.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB
joystick.vcxproj	2022/9/14 下午 03:47	VC++ Project	59 KB
joystick.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB
joystick_static.vcxproj	2022/9/14 下午 03:47	VC++ Project	60 KB
joystick_static.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB
keyboard.vcxproj	2022/9/14 下午 03:47	VC++ Project	59 KB
keyboard.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB
keyboard_static.vcxproj	2022/9/14 下午 03:47	VC++ Project	60 KB
keyboard_static.vcxproj.filters	2022/9/14 下午 03:47	VC++ Project Fil...	1 KB

21

Open Solution with Visual Studio

Make sure the building version matches your OS

22

Debug/Release Build

After finishing building, change to "Release" mode and build again

23

Examine the Built Binary Files

You can find the Debug/Release versions of *.lib (in the lib folder) and *.dll (in the bin folder), respectively

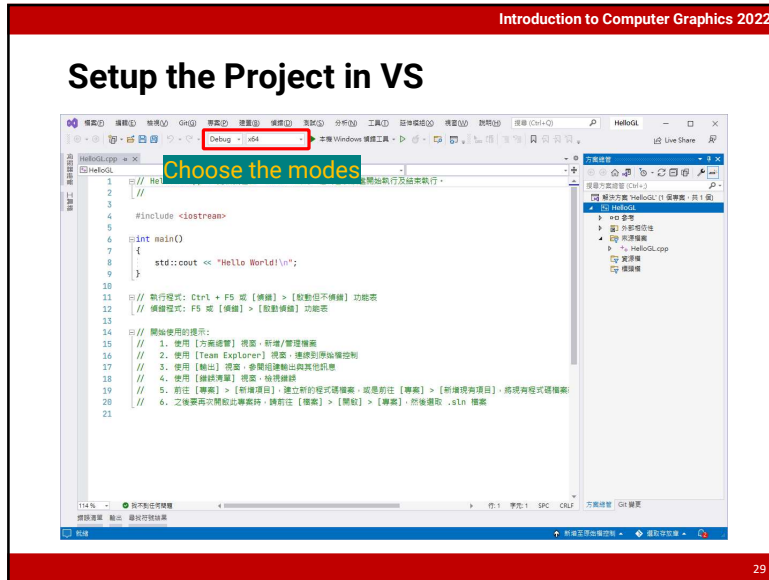
24

26

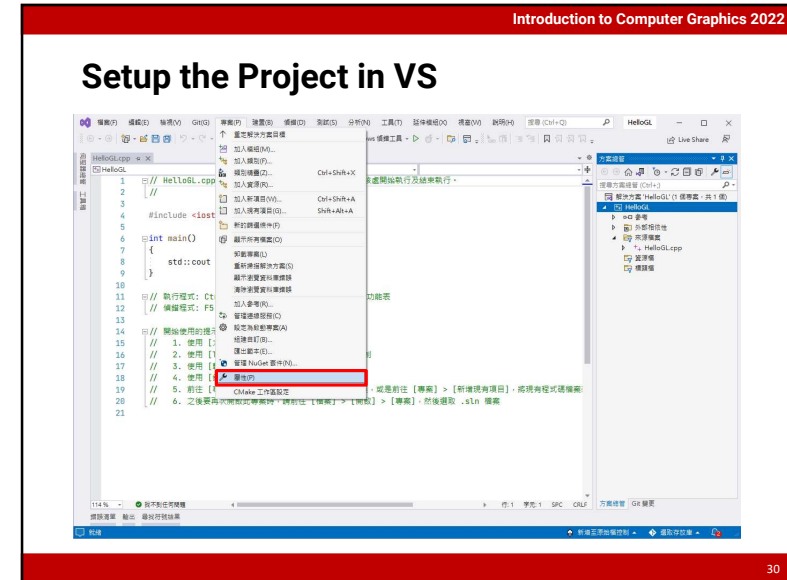
27

28

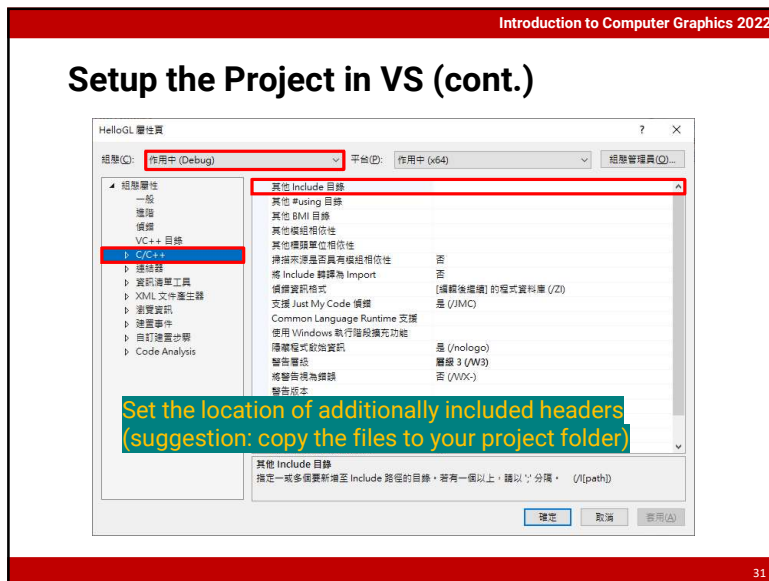
7



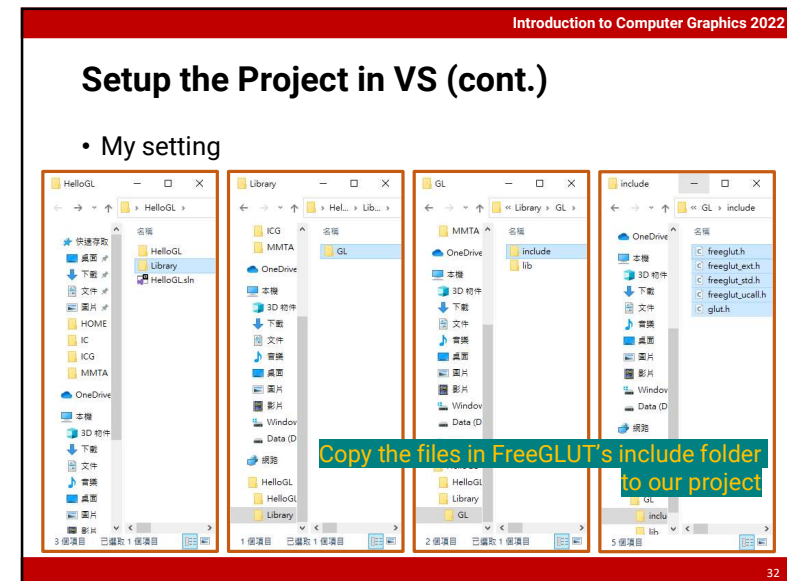
29



30

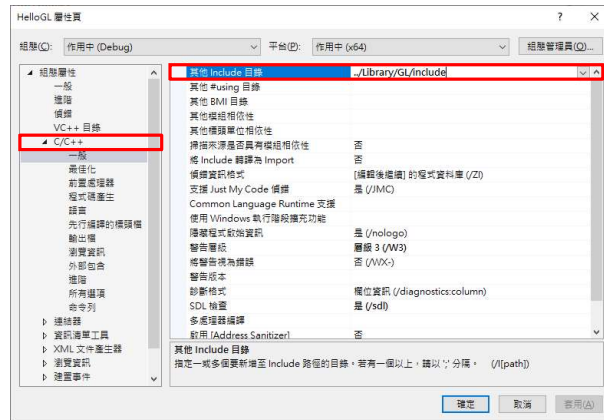


31



32

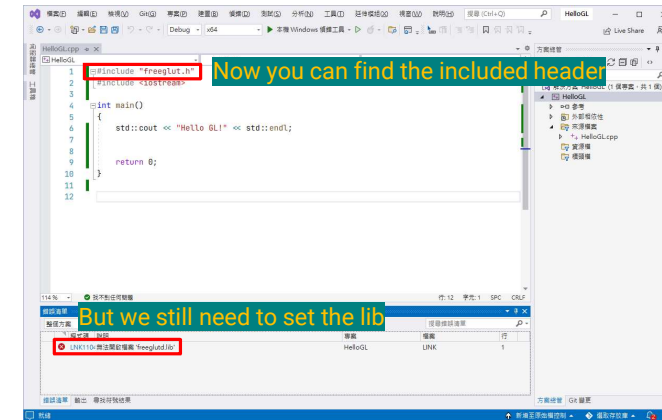
Setup the Project in VS (cont.)



33

33

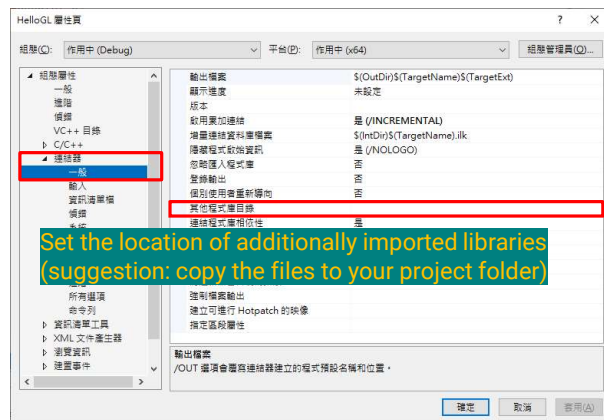
Setup the Project in VS (cont.)



34

34

Setup the Project in VS (cont.)

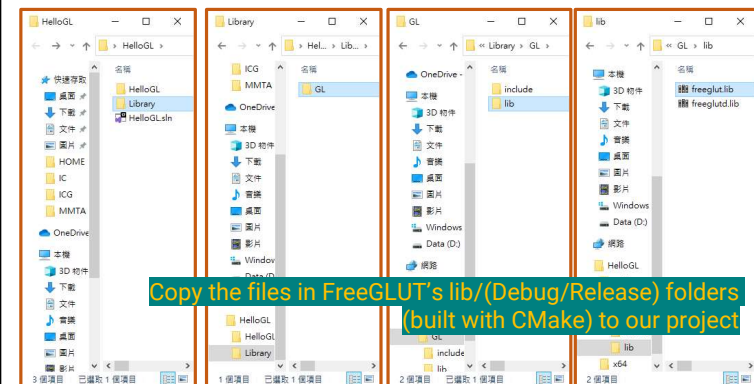


35

35

Setup the Project in VS (cont.)

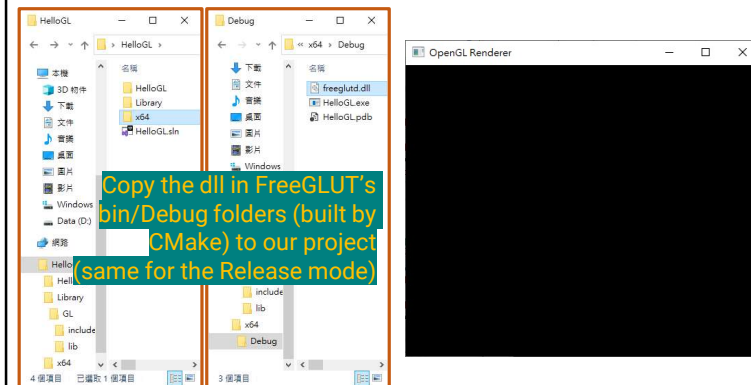
• My setting



36

36

Setup the Project in VS (cont.)



Any Questions?