OpenGL L1:
Setup and “Hello world”

3.1.2018

LUO Ziyun, Room 901, A7, Nanshan Park

What is OpenGL?

- OpenGL is a graphic API, but not a platform, more specifically a group
- Usually use C++
- Use graphics card manufacturers develop the libraries (NVidia, AMD, etc)

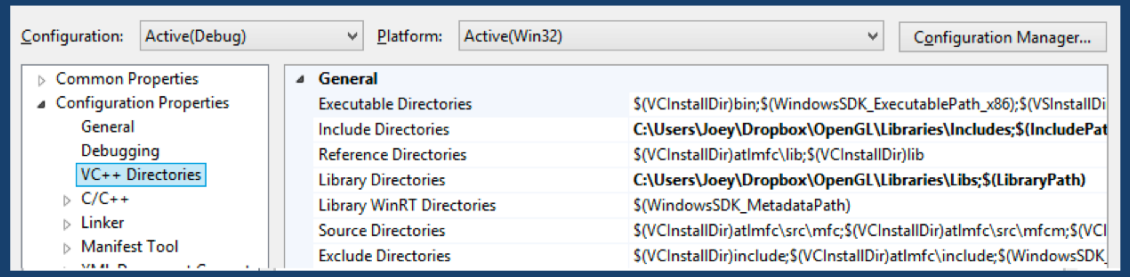
Task 1: Creating a window

- Create an OpenGL context and application window
- Some popular libraries: GLUT, GLFW, SFML, …

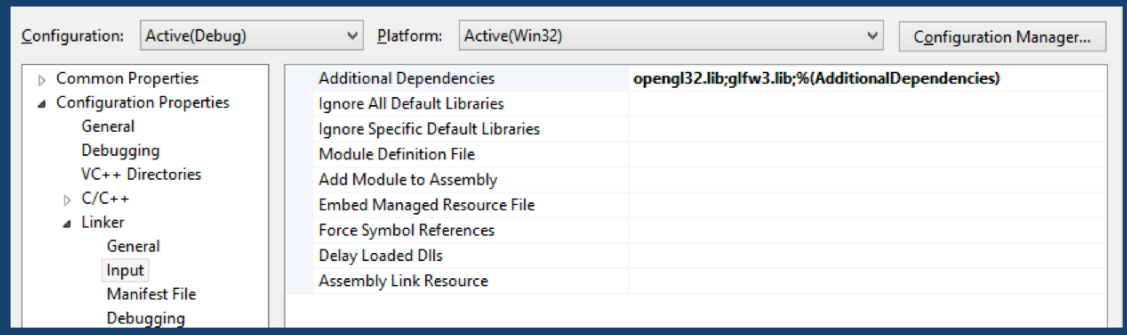
GLFW

- Download from official website
- Pre-built binaries (use 32 bit version) and header files vs. source code
- Use CMake to build solution
- Build in VS (on Windows), or gcc (on Linux)
- Find libraries and headers

Create a new project



- Link the libraries to the project
- Project Properties -> VC++ Directories -> Include Directories, Library Directories
- Project Properties -> Linker -> Input -> Additional Dependencies
- glfw3.lib, opengl32.lib
- Include <GLFW\glfw3.h>



GLAD

- Ease the process of finding functions location
- Go to the website, make sure select *core*, and generate a loader
- Get a zip, setup link, import gald.c to your project
- #include <glad/glad.h>

Code

```
int main()
{
    glfwInit();
    glfwWindowHint(GLFW_CONTEXT_VERSION_MAJOR, 3);
    glfwWindowHint(GLFW_CONTEXT_VERSION_MINOR, 3);
    glfwWindowHint(GLFW_OPENGL_PROFILE, GLFW_OPENGL_CORE_PROFILE);
    //glfwWindowHint(GLFW_OPENGL_FORWARD_COMPAT, GL_TRUE);

    return 0;
}
```

```
GLFWwindow* window = glfwCreateWindow(800, 600, "LearnOpenGL", NULL, NULL);
if (window == NULL)
{
    std::cout << "Failed to create GLFW window" << std::endl;
    glfwTerminate();
    return -1;
}
glfwMakeContextCurrent(window);
```

Code

GLAD

```
if (!gladLoadGLLoader((GLADloadproc)glfwGetProcAddress))
{
    std::cout << "Failed to initialize GLAD" << std::endl;
    return -1;
}
```

Viewport

```
glViewport(0, 0, 800, 600);
```

```
void framebuffer size callback(GLFWwindow* window, int width, int height);

void framebuffer_size_callback(GLFWwindow* window, int width, int height)
{
    glViewport(0, 0, width, height);
}
```

```
glfwSetFramebufferSizeCallback(window, framebuffer_size_callback);
```

Code

```
while(!glfwWindowShouldClose(window))
{
    glfwSwapBuffers(window);
    glfwPollEvents();
}
```

Render loop

```
glfwTerminate();
return 0;
```

Release resource

Assignments

- Create a window whose title is your name and student #
- Change the window's background color
- write something in the window
- Screenshot result with your pc background wallpaper