CS171 Warm-up Assignment: Programming Simple Graphics Program with OpenGL

Introduction

In this assignment, you are required to create your first simple graphics program using OpenGL. As you have learned in the course as well as the tutorials, to start up your first graphics program, you will need to use GLFW to create a window. By default, GLFW will create a window with double buffering enabled. Then, you can use the basic OpenGL calls to draw 2D/3D objects on the window.

In the following, we will give you the specifics about what you need to accomplish, as well as some related guidelines in order to assist your programming.

Note

Before doing the assignment, please read the materials on OpenGL programming at site1, site2. You can also read site3 which conducts OpenGL rendering based on OpenGL Mathematics (GLM) library.

Programming Requirements

- You are required to draw any 2D/3D objects as you like based on the OpenGL window created for you.

Submission

You are required to submit the following things through GitHub repository:

- project scripts and an executable program in Coding folder.
- a PDF-formatted report which describes what you have hone in Report folder.

Submission deadline: 22:00, Sep 24, 2021

Grading rules

Take it easy! This is a warm-up assignment and we will not grade your submissions. Please relax and have fun in programming with OpenGL! :D

Skeleton Project/Report Template

The skeleton program and report template will be provided once you accept the assignment link of GitHub classroom which we published in Piazza. If you accept the assignment through link properly, a repository which contains the skeleton project and report template will be created under your GitHub account. Please follow the template to prepare your report.

You should complete your assignment submission to your repository through GitHub before the deadline.

Implementation Guide

Git Classroom

Accept the assignment in this link or download the zip_package to start your assignment.

1.Environment Setup

In our tutorial class, we will explain how to set up a CMake environment with our skeleton program. You need CMake to build your code. In "libs" directory we provide two libraries GLFW and GLEW. GLEW is an OpenGL extension wrapper library that contains all the OpenGL API functions based on which you can call to make OpenGL calls, whereas GLFW is a multi-platform library for creating windows on which your objects are drawn.

To build the project, firstly you need to download CMake if you do not have one. Then run command

```
mkdir build
cd build
cmake ..
cmake --build
```

These commands first make a directory whose name is "build" and then jump into it. After that, it uses CMake to configure the project and builds the project.

Besides, we recommend using Visual Studio in Windows and Visual Studio Code in Linux. Both of them can build the CMake-based projects automatically (maybe with help of some plugins). If you are an experienced developer, you can choose whatever you like.

2. Creating the window program using GLFW

Once you have set up your environment, you can start creating your first window program using GLFW. We have already created a window for you in the skeleton. You can take a reference from this tutorial which illustrates the step-by-step process for window creation.

You can do some experiments with the parameters in window creation, like window width, and see what happens.

3. Drawing primitives in OpenGL

There are a couple of methods to draw primitives in OpenGL and you can refer to this site for a brief introduction of each method.

One method, called Immediate Mode using glBegin and glEnd, provides a neat and simple way of drawing 3D primitives. You can follow this site for a more detailed description. However, the glBegin/glEnd combo has unfortunately become obsolete, and it is now superseded by some modern standards. It is useful to get familiar with the modern OpenGL rendering technique using VAO and VBO. This tutorial offers a comprehensive illustration on how to draw a triangle in a modern OpenGL style.

For this assignment, you can complete the code in "main.cpp" with any method of them. But we do recommend the modern style with VAO and VBO.

4. Showcase

After the above procedures, you will see your masterpiece :D in the OpenGL window. Here we provide an 'A' as an example.

Do whatever you want with OpenGL

Up to now, you have already known how to draw basic primitives in OpenGL. Then, you can do some more interesting and advanced things.

- You can change the objects' positions/colors with time (function `glfwGetTime` may help).
- You can use the keyboard to move an object (function `glfwGetKey` may help)..
- You can draw some 3D objects.
- ...

Use your imagination and draw some beautiful scenes!