
Homework 2 - GUI and Draw simple graphics

deadline: 3.12 周二 23:59前

Introduction

在这次作业里，恭喜大家终于可以动手编写你们的第一个OpenGL程序了！

所谓“工欲善其事，必先利其器”，在编写OpenGL程序之前，我们希望大家先写一个简单的小工具——一个在后面所有作业都将用到的GUI界面。这个GUI要求是可交互的，当然也希望尽量美观、简洁，方便大家后续的使用。

完成了GUI工具后，大家需要使用OpenGL的知识，绘制CG世界里最基本的图元（点、线、面等）

作业严禁抄袭，被发现者当次作业0分！

References

1. OpenGL相关知识

本课程要求大家使用现代OpenGL核心模式进行编程开发，也就是说OpenGL版本须为**OpenGL 3.3及以上版本**。目前OpenGL官方最新版本为4.6，但是还是鼓励大家从学习OpenGL 3.3开始，因为所有OpenGL更高的版本都是基于3.3的基础上，引入了额外的功能，核心的架构没有改动。

另外，在我们作业中，要求大家统一使用**GLFW或者freeglut**，不要使用glut（已经过时）。

推荐一个规范学习OpenGL的教程网站（基于GLFW）：[OpenGL教程](#)，里面的教程十分详细，最重要是提供完整代码，而且代码特别规范。希望大家能够在课程开始阶段就注意编程规范，好的编程规范能避免在后面走弯路。另外，由于此教程网站提供完整代码，且大部分代码会我们课程作业重合，所以希望大家仅把教程里的代码作为参考，去理解每句代码背后的意义以及现代OpenGL的相关概念，不可以直接把代码复制上交（TA们会严查此类抄袭）。

1.1 OpenGL环境配置

不同操作系统下OpenGL环境配置有差别，但本质上还是头文件(.h)的添加以及库文件(.lib)的链接。下面推荐一个规范且详细的Windows下针对Visual Studio的使用GLFW编程的OpenGL环境配置：

- [GLFW环境构建](#)（同样来自上面的learnopengl教程网站）

网上还有很多其他的教程。大家先自行配置，有问题再提出。

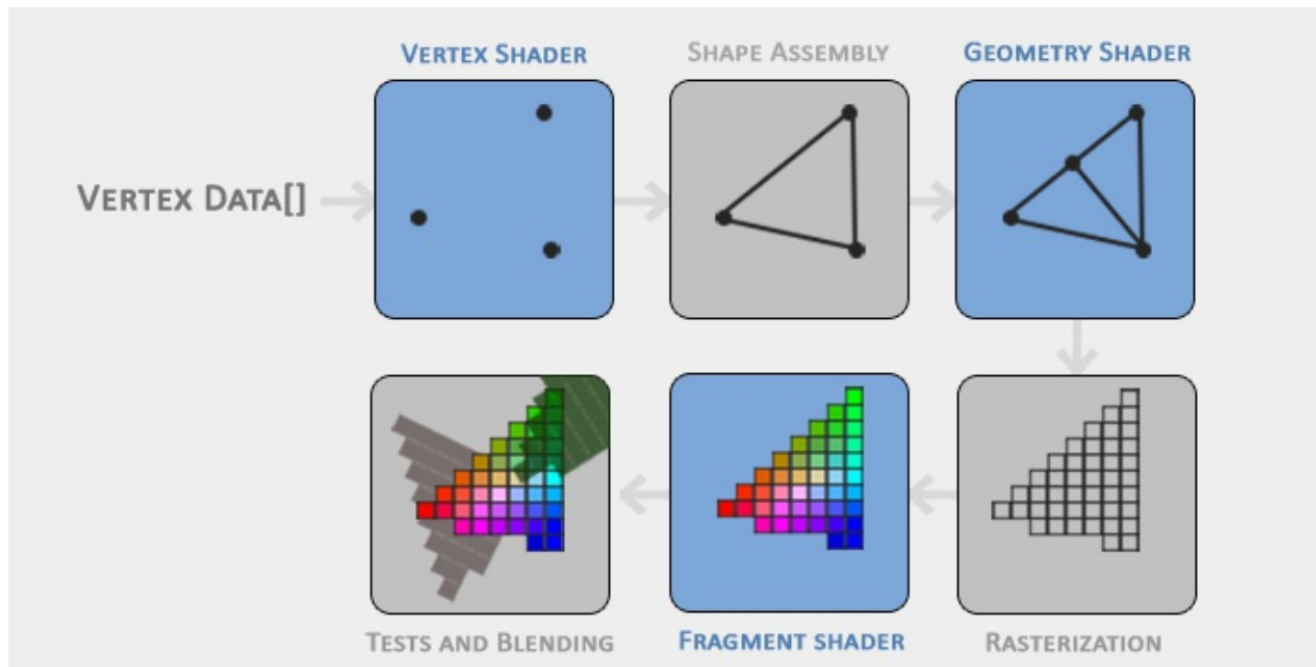
1.2 GLFW和freeglut

这两个都是用来帮助我们管理窗口、读取输入、处理事件。GLFW相比于freeglut更加轻量级。

1.3 OpenGL图形渲染管线(Graphics Pipeline)

在进行OpenGL编程前，我们要对OpenGL的图形渲染管线有个初步的了解。

在OpenGL中，物体都是在3维空间里的。但是，计算机显示器上显示的都是2维的像素阵列。将物体从3维空间映射到2维平面上，就是OpenGL的一个重要工作。这也就是OpenGL图形渲染管线所要做的。下图为OpenGL图形渲染管线的简化流程图：



在本次作业里，大家虽然是刚开始使用OpenGL编程，但是已经要接触到大一堆OpenGL相关的知识了。在这个过程中可能大家会对某些概念很难以立刻理解，don't worry! 这种情况下大家可以暂时跳过，有个印象即可。很多概念老师会在后面的课上讲到，而且随着后面使用OpenGL越来越多，大家会渐渐能理解这些背后的概念的。

2. GUI介绍

2.1 Qt

(不建议大家使用Qt !!!)

Qt是一款C/C++语言下的图形界面编程工具。但是要是大家使用GLFW编程的话，在Qt下，很多函数方法和编程逻辑都与传统的GLFW有很大区别！比如GLFW常用的 `glUseProgram()`，在Qt下变成了 `program.bind()`，详见下面这篇帖子，里面有对两者的多个常用function的写法做了对比：

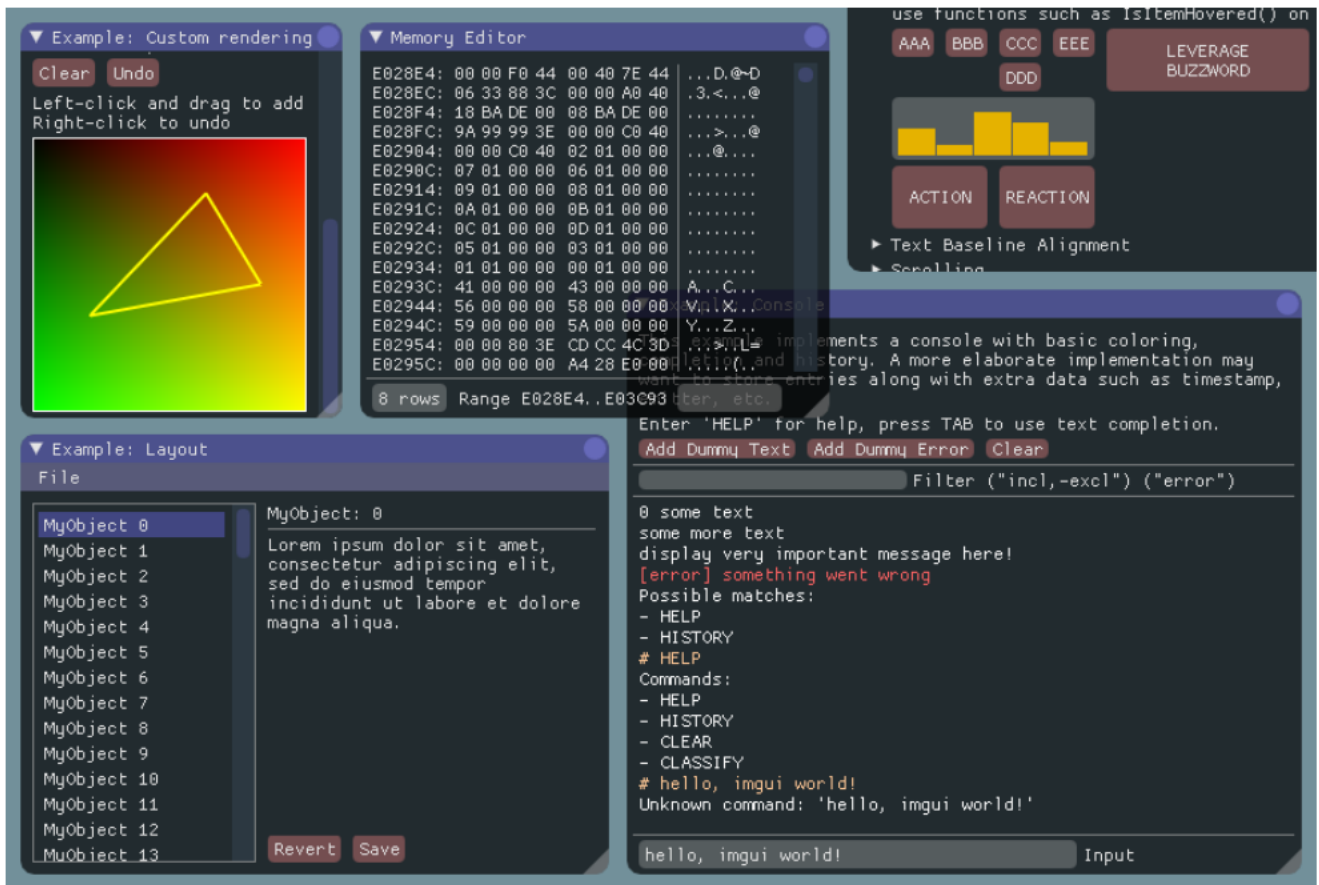
- [Qt下使用Shader绘制三角形](#)

另外，使用Qt，提交的作业会很大！所以还是不建议大家使用Qt，而是使用下面这个使用简单且美观的GUI工具！

2.2 ImGui

ImGui是一个c++图形用户界面库。他的结构简单使用成本低，并且功能强大可以满足大部分图形学编程所需的界面需求。其适用于集成在实时3D应用,全屏应用,嵌入式应用程序,游戏等多种程序。

和Qt比起来ImGui最大的优点是他的主要结构是几个头文件，不要繁杂的安装过程，直接将对应头文件导入到程序中即可使用，而且上手比较简单。



详细参考ImGui的github: [ImGui Github](https://github.com/ocornut/imgui)

Homework

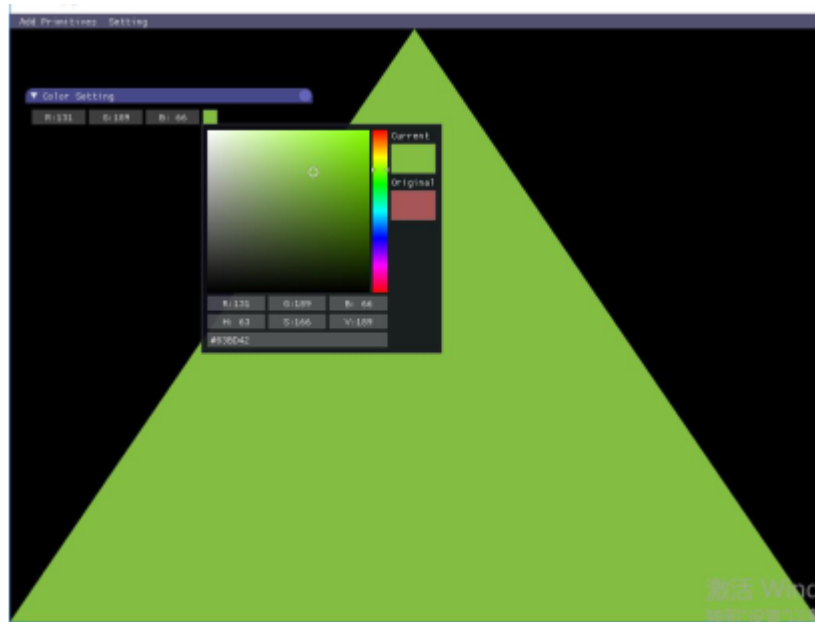
虽然learnopengl教程网站有很多现成的代码, 但是希望大家全部手打, 而不是直接copy。

Basic:

1. 使用OpenGL(3.3及以上)+GLFW或freeglut画一个简单的三角形。
2. 对三角形的三个顶点分别改为红绿蓝, 像下面这样。并解释为什么会出现这样的结果。



3. 给上述工作添加一个GUI，里面有一个菜单栏，使得可以选择并改变三角形的颜色。



Bonus:

1. 绘制其他的图元，除了三角形，还有点、线等。
2. 使用EBO(Element Buffer Object)绘制多个三角形。

作业要求:

1. 把运行结果截图贴到报告里，并回答作业里提出的问题。
2. 报告里简要说明实现思路，以及主要function/algorithm的解释。