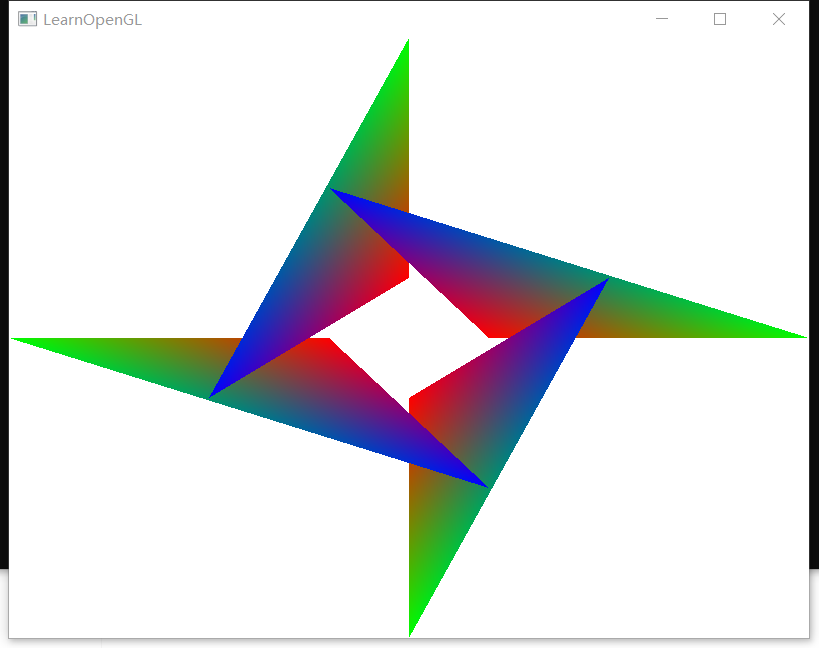
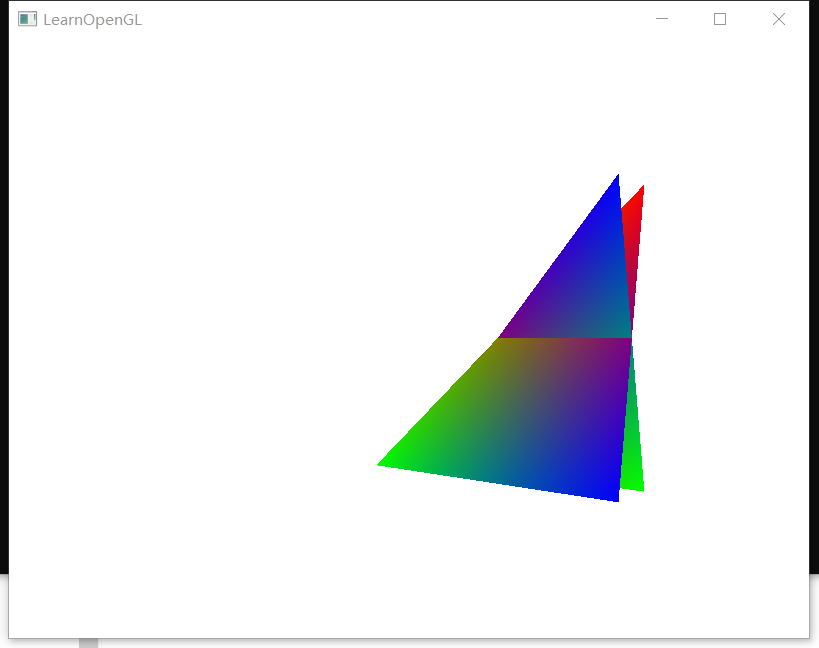
第二次作业作业报告

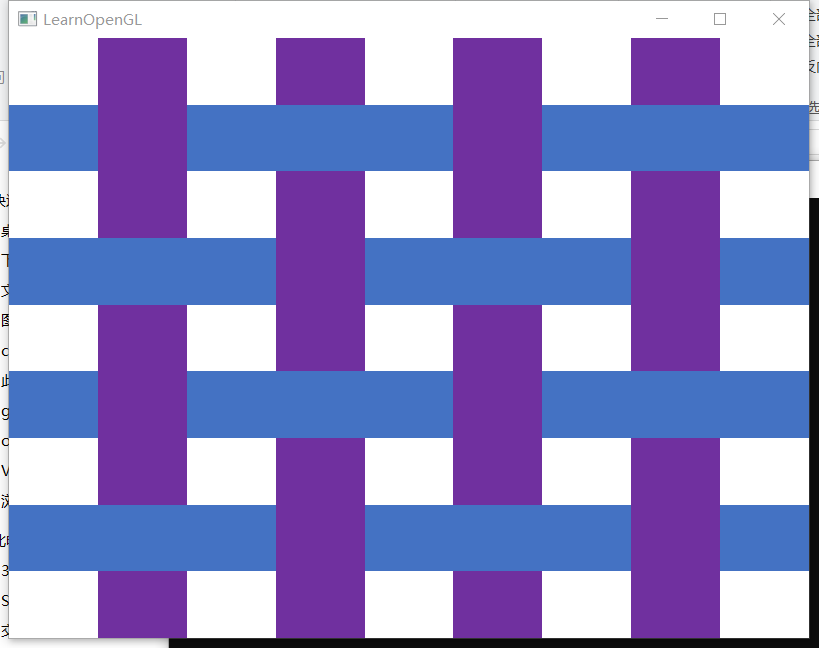
1. 环境搭建
   1. Windows10+Visual Studio
   2. 使用了GLFW，参考教程加入了glad（指针管理）跟glm（矩阵运算）
2. 具体工作
   1. 配置环境，上次作业在ubuntu上编写，本次决定在windows上编写，于是又配了一遍环境（作了大死）。
   2. 学习网上opengl教程，了解VAO、VBO的使用，着色器的使用，深度测试的使用，以及根据矩阵变换实现视角的变换
   3. 编写作业代码
   4. 整理课程笔记
   5. 编写作业报告
3. 技术方案
   1. 如何使用VAO、VBO储存顶点的属性值
      1. 首先对顶点作归一化处理（图像可以随窗口大小变化）
      2. 将顶点数据存放在float数组vertex中
      3. 为VAO、VBO申请ID
      4. 将VAO与VBO与数组vertex绑定
      5. 使用glVertexAttribPointer（）解释顶点
      6. 使用时绑定VAO
   2. 如何使用着色器
      1. 编写着色器程序
         1. 使用opengl内置语法编写了简单的着色器
         2. 编写了bertexShader与fragmentShader
      2. 编译并链接着色器程序得到shaderProgram
      3. 使用着色器程序glUseProgram(shaderProgram);
   3. 如何开启深度测试
      1. glEnable(GL\_DEPTH\_TEST);
   4. 如何自己实现扫描线算法
      1. 如何计算当前扫描线与多边形的交点
         1. 使用活动边表
      2. 如何处理极值顶点
         1. 当作两个点来用
      3. 如何快速配对交点并着色
         1. 便利活动边表
   5. 如何编写分段函数
      1. 使用了比较蠢的办法，将原图像划分为多个小块（72个矩形），为每个小块设置深度（深度值为x、y的函数），分别绘制每个小块。
4. 运行效果
   1. Task1



* 1. Task2



* 1. Task3



1. 操作说明
   1. RELEASE运行
      1. 直接在REALEASE文件夹中运行三个task.exe，相关依赖已在相同路径下
   2. VS工程运行
      1. 使用VS打开opengl\_hw2/gl\_test.sln
      2. 由于只使用了一个工程，所以3个文件无法同时运行，在每个文件中都包含main（）函数，（为了可以保证运行时只有一个main（），已改为了main111（）等名称），运行时将对应文件中的main（）改回原名称即可。
2. 参考资料
   1. Learn OpenGL CN <https://learnopengl-cn.github.io/>
   2. CSDN论坛n个网站<https://www.csdn.net/>
   3. 简书n个网站<https://www.jianshu.com>
   4. 特别感谢谷歌提供的高质量搜索<https://www.google.com.hk/>