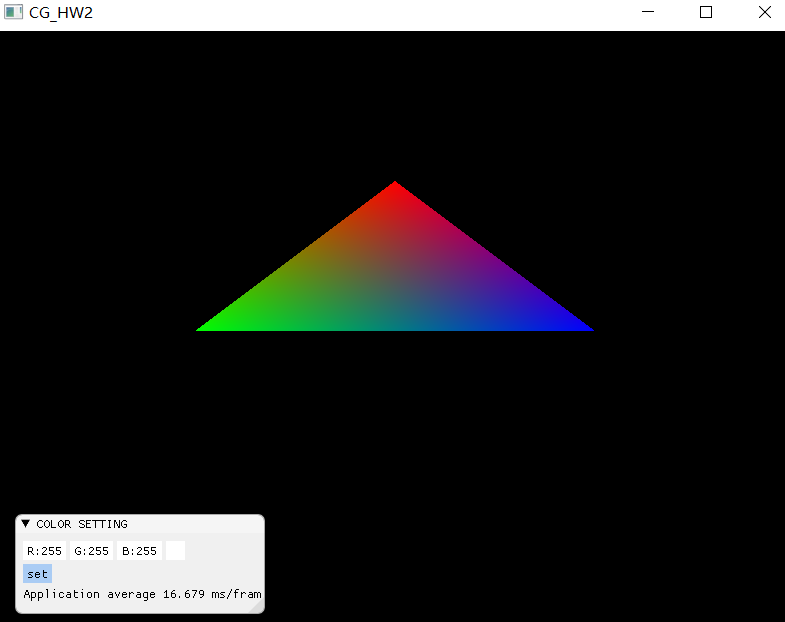
# BASIC：

1. 使用OpenGL(3.3及以上)+GLFW或freeglut画一个简单的三角形。



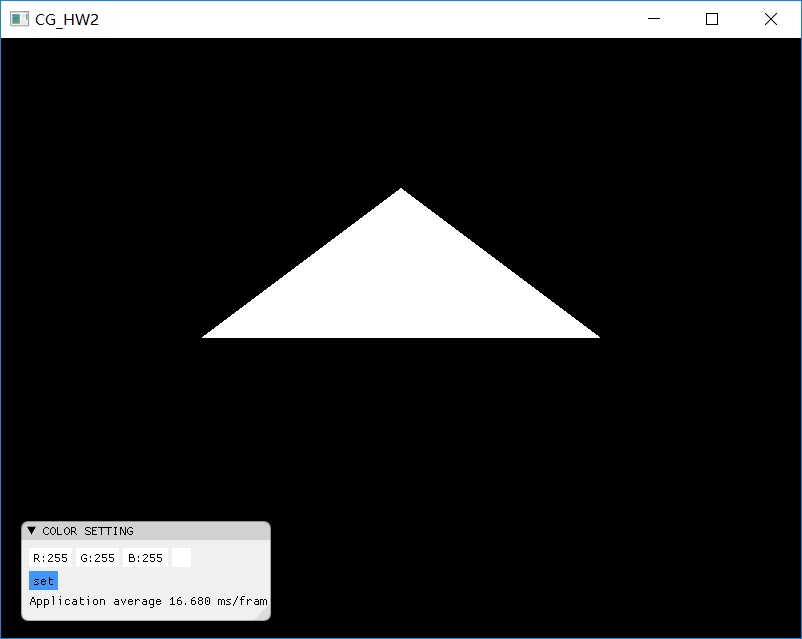
1. 对三角形的三个顶点分别改为红绿蓝，像下面这样。并解释为什么会出现这样的结果。

这是因为在片段着色器中会进行的片段插值。当渲染一个三角形时，光栅化阶段会生成比原指定顶点更多的片段。光栅会根据每个片段在三角形形状上所处相对位置，插值所有片段着色器的输入变量。

即是光栅化的时候，根据我们提供的3个顶点来确定三角形边和内部的像素点。因为只有3个顶点有颜色输入变量，OpenGL会将其他像素点进行按与3个顶点的距离等比例插值作为输入，所以结果与调色板类似。

1. 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。

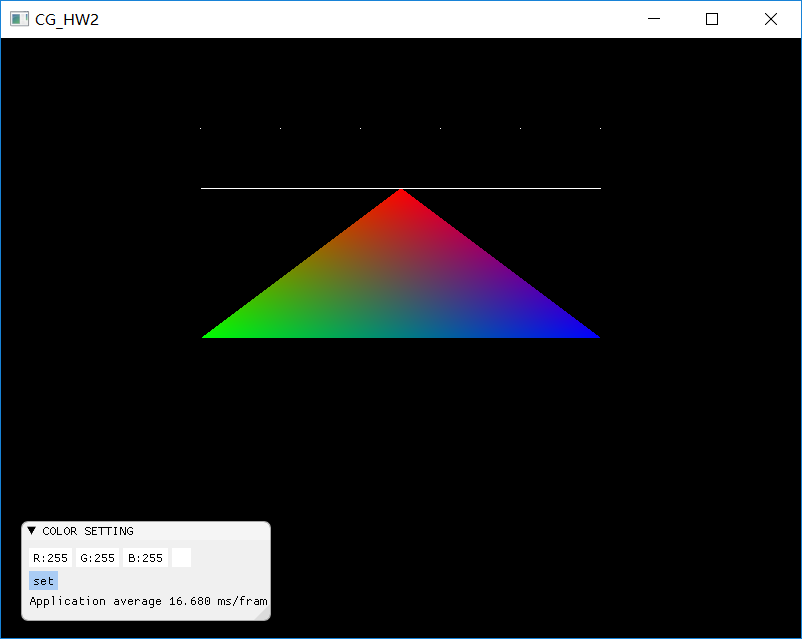
（选择颜色值后按SET按钮即可生效）



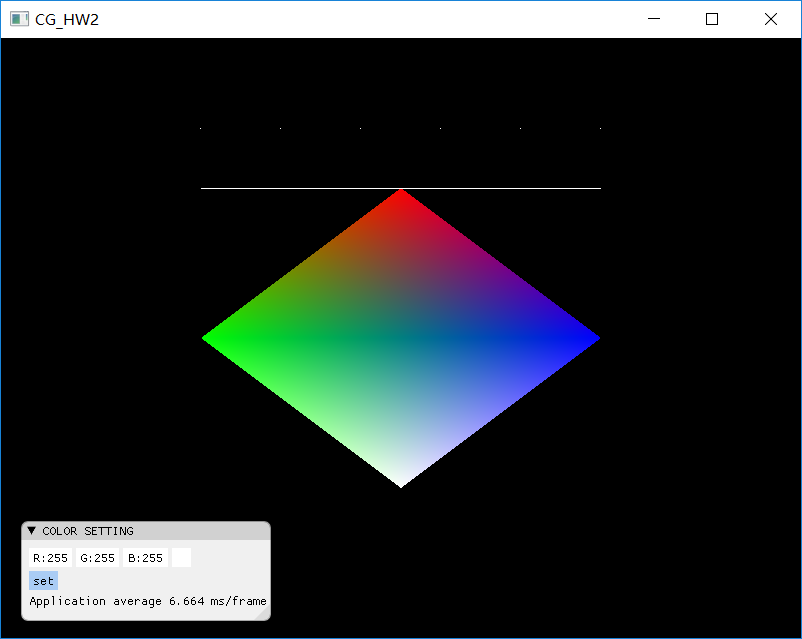
# Bonus

1. 绘制其他的图元，除了三角形，还有点、线等。

(三角形上面一条白色直线，直线上面有五个白点)



1. 使用EBO(Element Buﬀer Object)绘制多个三角形。

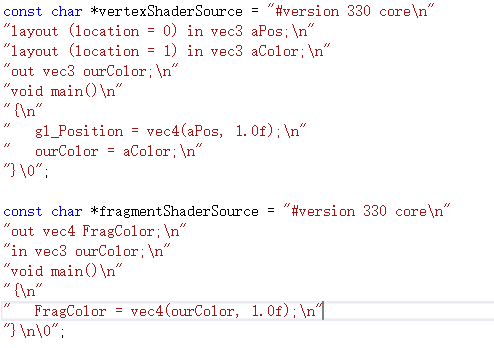


# 代码思路解析

附代码文件main.cpp

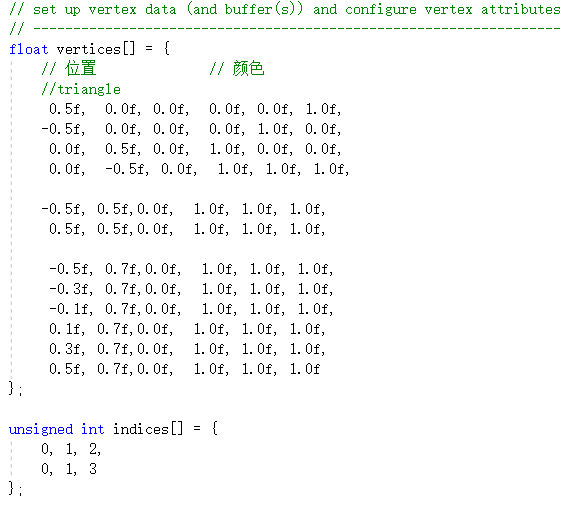
1. 两个着色器代码

顶点着色器主要对传入的顶点数据预处理，传递位置和颜色值，片段着色器则对颜色值作最后的处理。



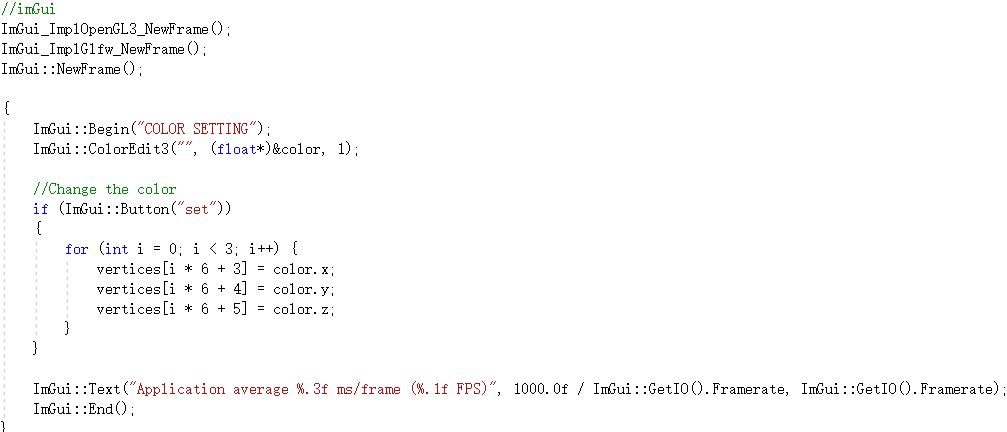
1. 顶点数据设置

四个顶点画两个三角形，两点画直线，五个离散点。



1. Imgui控件

当用户选择颜色控件并点击SET按钮时，更新顶点数组内的颜色值，并重新绑定输送数据到显卡。



1. 渲染核心部分

