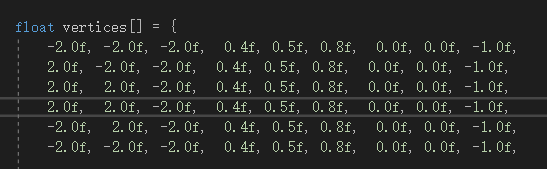
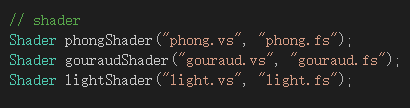
**Basic:**

1. 绘制一个cube，vertex的数据格式如下所示：



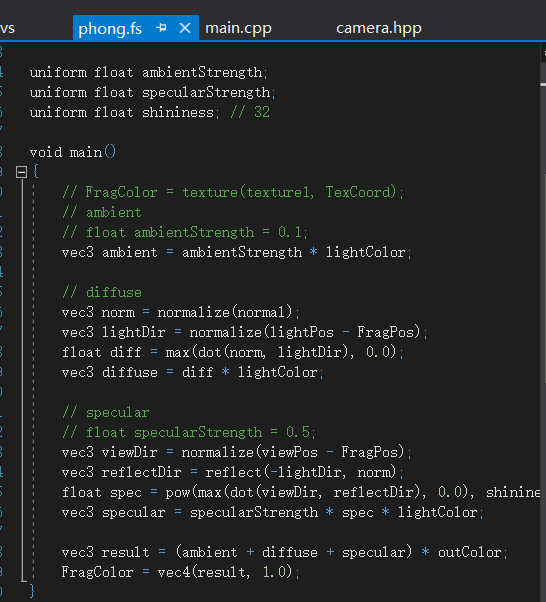
前三个为顶点坐标，中间3个为面的颜色（本作业中忽略掉），最后为法向量；

2.分别实现Phong Shader和Gouraud Shader：



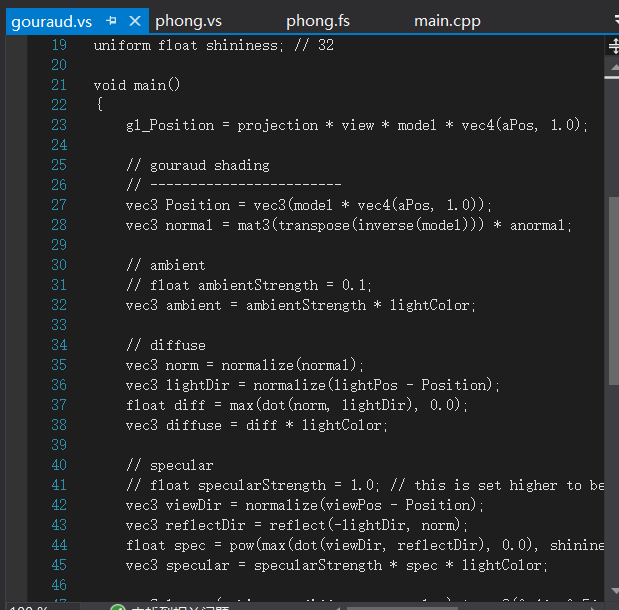
**Phong Shading：**

按照教程中的写法完成vs和fs文件即可，其中，phongShading是在frag着色器中进行光照颜色的计算的：



**Gouraud Shading：**

Gouraud Shading的颜色计算发生在顶点着色器中：



计算完color之后传给frag着色器：

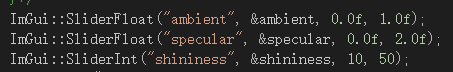


**实现原理：**

Gouraud shading是先计算顶点的颜色，然后根据顶点信息，对中间的颜色进行二维的插值。因此其代码直接在顶点着色器中实现。

Phong shading对每个片段（fragment）计算光照，点的法向量通过顶点的法向量插值得到。因此其代码在片段着色器中实现，其开销会比Gouraud要大。

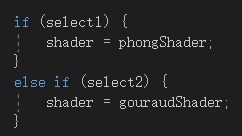
3. 设置GUI以修改参数：



因为diffuse是计算求出来的，因此无需对其进行改变调节。

用一个引用来切换两种Shader：



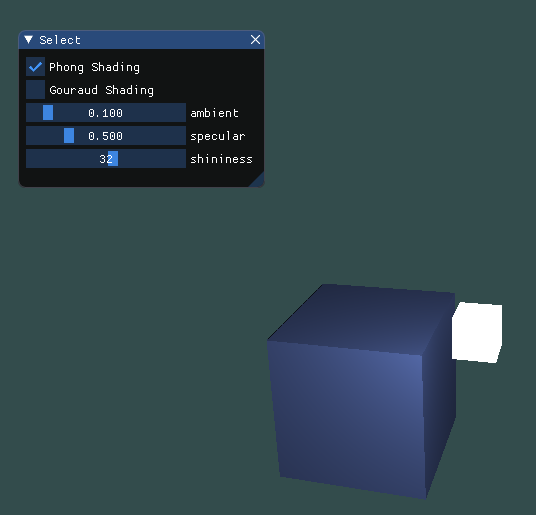


**Bonus:**

更改lightPos的值来使光源移动，使其绕中间的cube旋转：



**结果截图：**



通过GUI可以切换两种shading，并且调节参数。