**0 上一章节我们创建了一个黑客帝国风格的材质效果。本章节我们将把光源的种类进行扩充。**

（1）定向光实现。

（2）点光源实现。

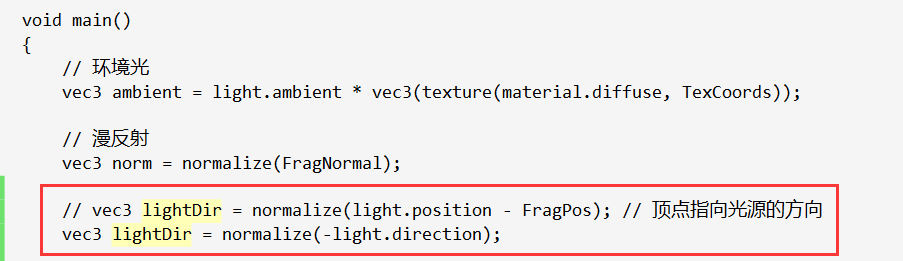
（3）聚光实现。

（4）将不同种类的投光物进行抽象。

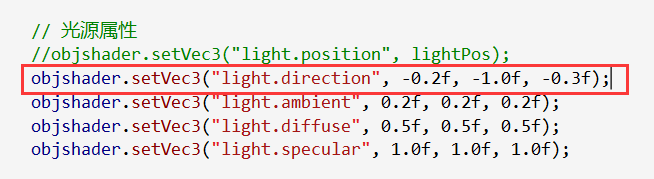
**1 定向光实现。**

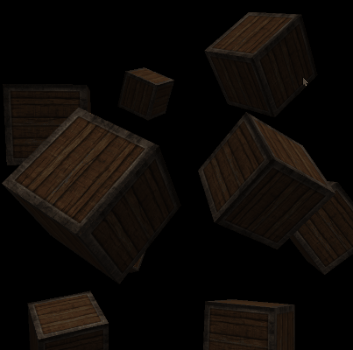
1.0由于定向光的光照方向不会随着光源位置的移动而发生变化，所以光源属性里不需要光源的位置，只需要光源的方向即可。





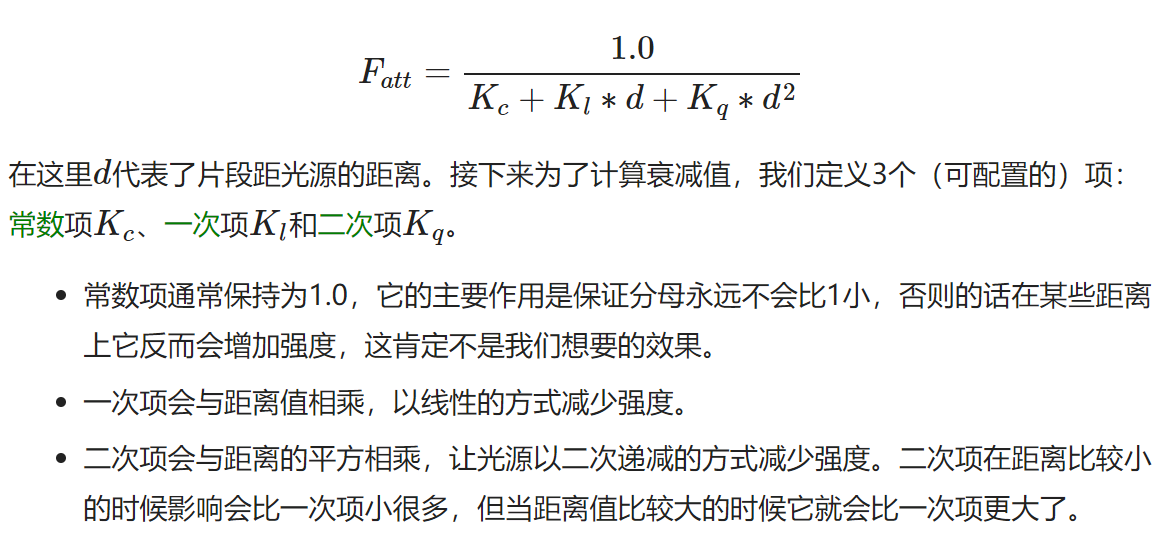
1.1 设置好定向光的方向向量，编译运行程序可得到固定光照方向的场景。

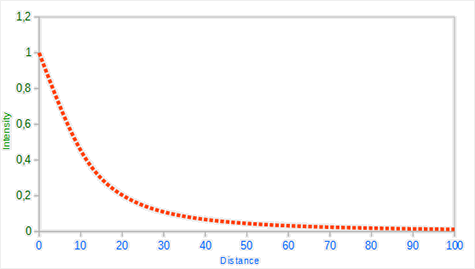




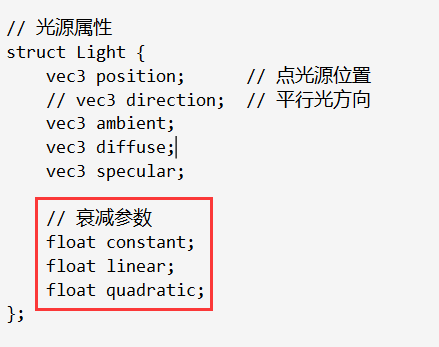
**2 点光源实现。**

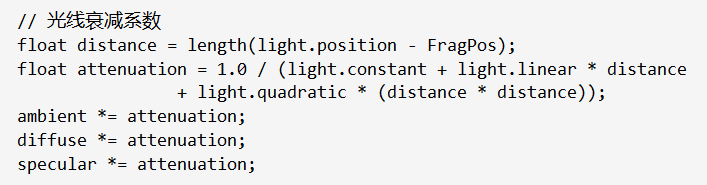
2.0 点光源的光源方向根据光源的位置确定。点光源的重点在于从该光源发出的光照强度会随着光源到着色片段的距离增加而逐渐减弱。光源离片段越近光越亮，离的越远光越暗。具体的衰减关系如下所示：





2.1 我们根据点光源的衰减公式，设置好点光源的光照属性，并使用这些衰弱参数实现光照衰弱公式计算出光线的衰减系数，将衰减系数乘以Phong光照模型参数便是点光源的光照模型。



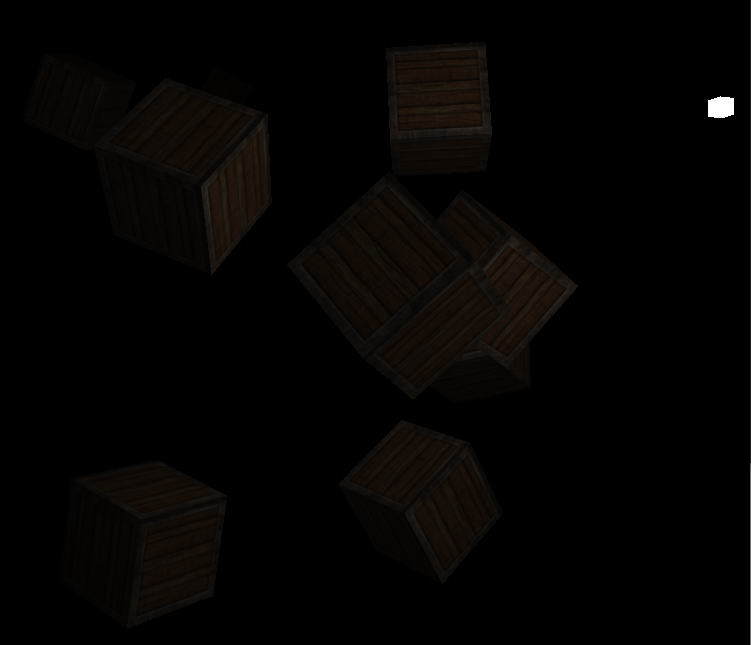


2.2 点光源的衰减系数一般情况下可按下表进行经验值设置。



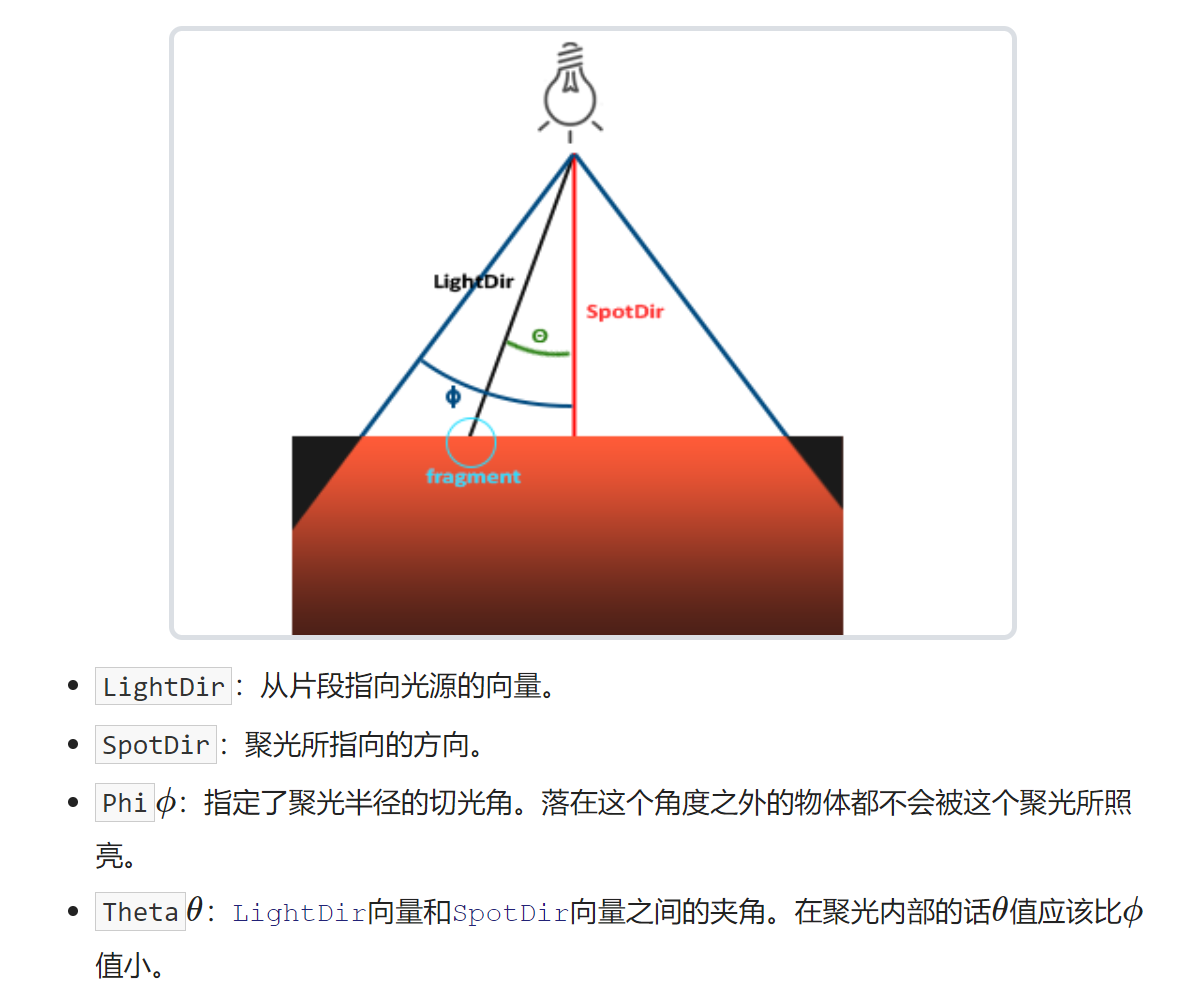


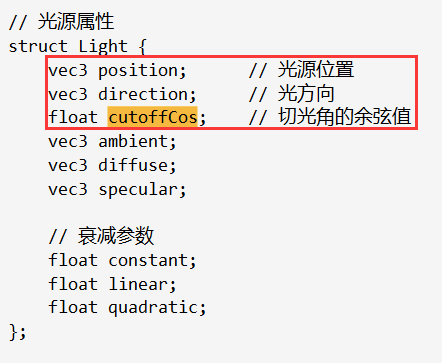
2.3 编译运行程序，可得到点光源的渲染场景。此时可以看到由于点光源距离被照物体较远，所以整个场景看起来很暗。

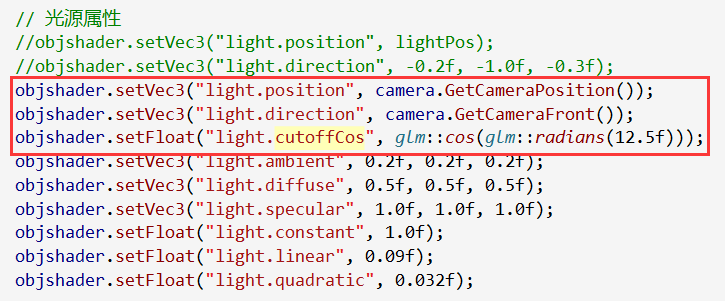


**3 聚光实现。**

3.0 聚光的原理关键在于需要预先设置好光锥的切光角度，根据着色片段与光源的位置来判断该着色片段是否处于光锥范围之内。如果在光锥范围内则进行着色，如果不在光锥范围内则只对该片段进行环境光的渲染。



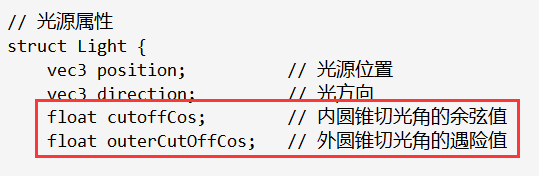


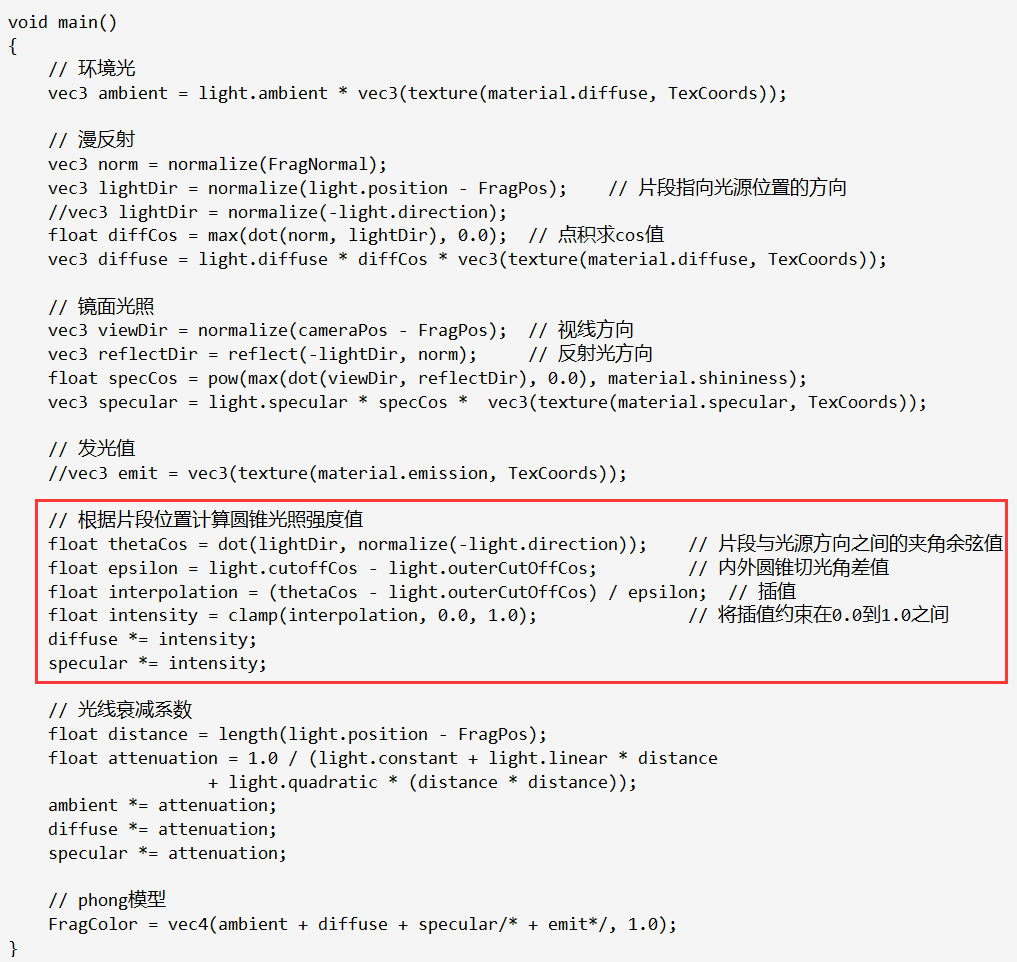


3.1 编译运行程序后可看到聚光源的效果，这个效果就像手电筒的一束光照射在箱子上一样。但是此时光锥边缘的片段光照过渡非常生硬，在光锥之内的片段很亮，在光锥之外的片段就直接黑掉了。



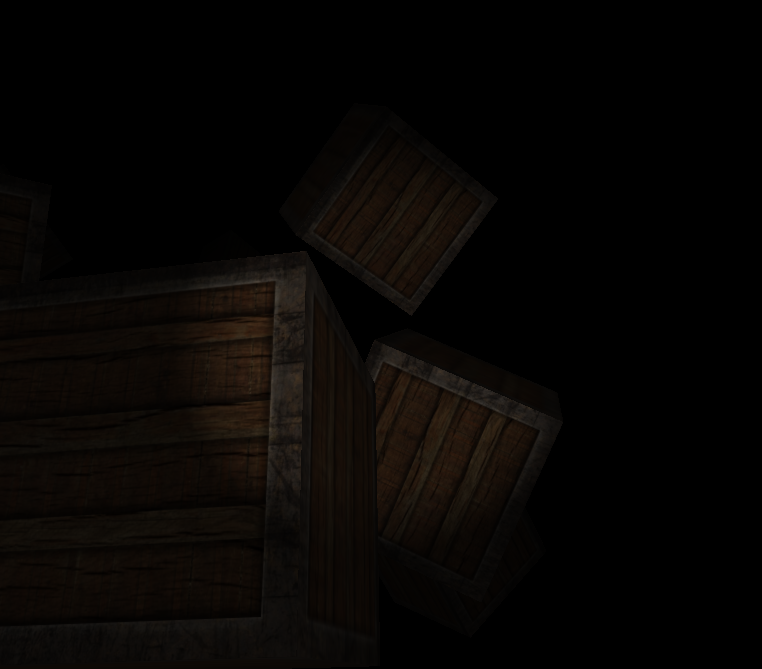
3.2 为了解决光锥边缘光线过渡生硬的问题，我们需要引入内外光锥的概念。此时如着色片段在内光锥之内则会很亮。如着色片段在内光锥与外光锥之间则会根据位置进行插值计算，光照强度由内至外会逐渐减少。当着色片段位于外光锥之外时，则只进行环境光渲染。





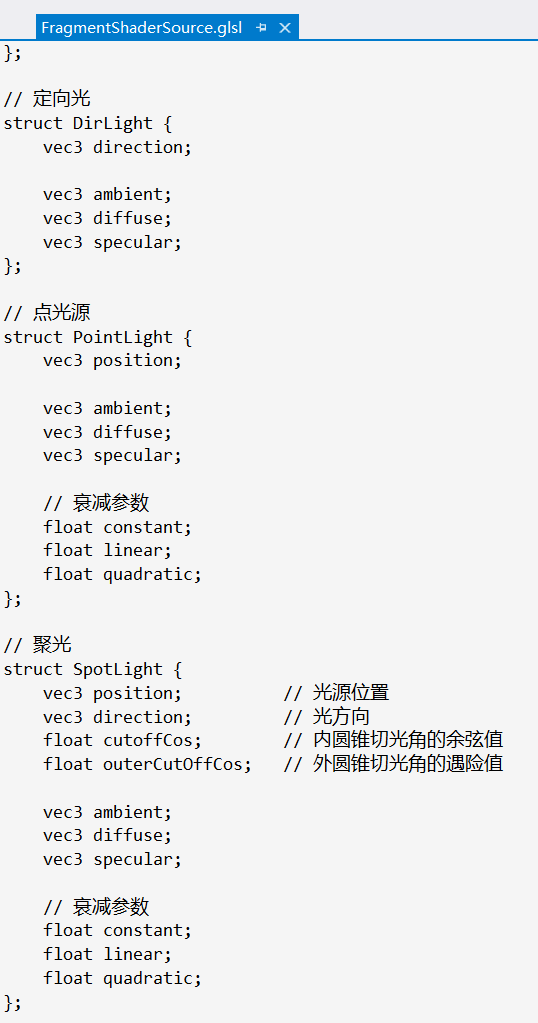


3.3 编译运行程序可得到一个光锥边缘被过渡的场景。此时的场景是不是很有恐怖片里主角拿着手电筒去鬼屋场景探索的氛围？

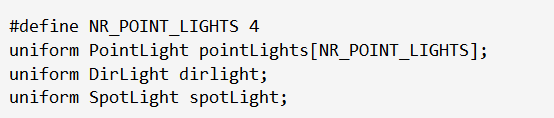


**4 将不同种类的投光物进行抽象。**

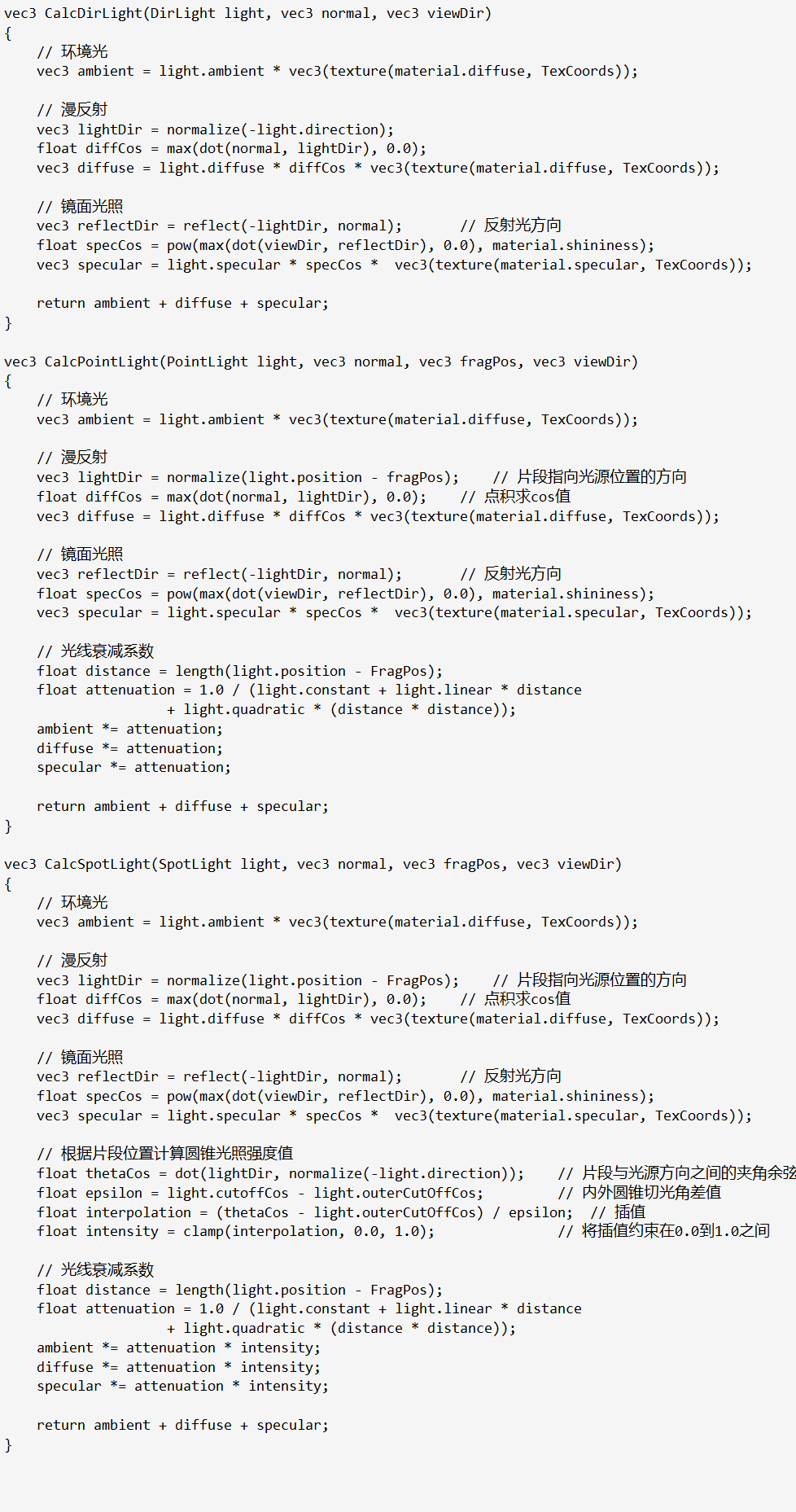
4.0 我们已经将三种不同类型的投光物给实现了一遍，现在我们需要将其抽象整理出来。首先我们需要把三种投光物的光照属性参数抽象成对应的光照属性数据结构。



4.1 定义好投光物的光照属性数据结构之后，我们需要使用这些数据结构来生成投光物的对象。（其中点光源对象是一个数组，意味着我们可以在场景中设置多个点光源）



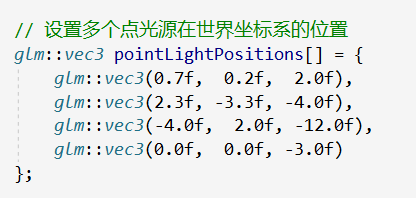
4.2 定义三种投光物各自对应的光照计算函数接口。



4.3 此时的Phong着色模型将被抽象为三种投光物的着色信息之和。

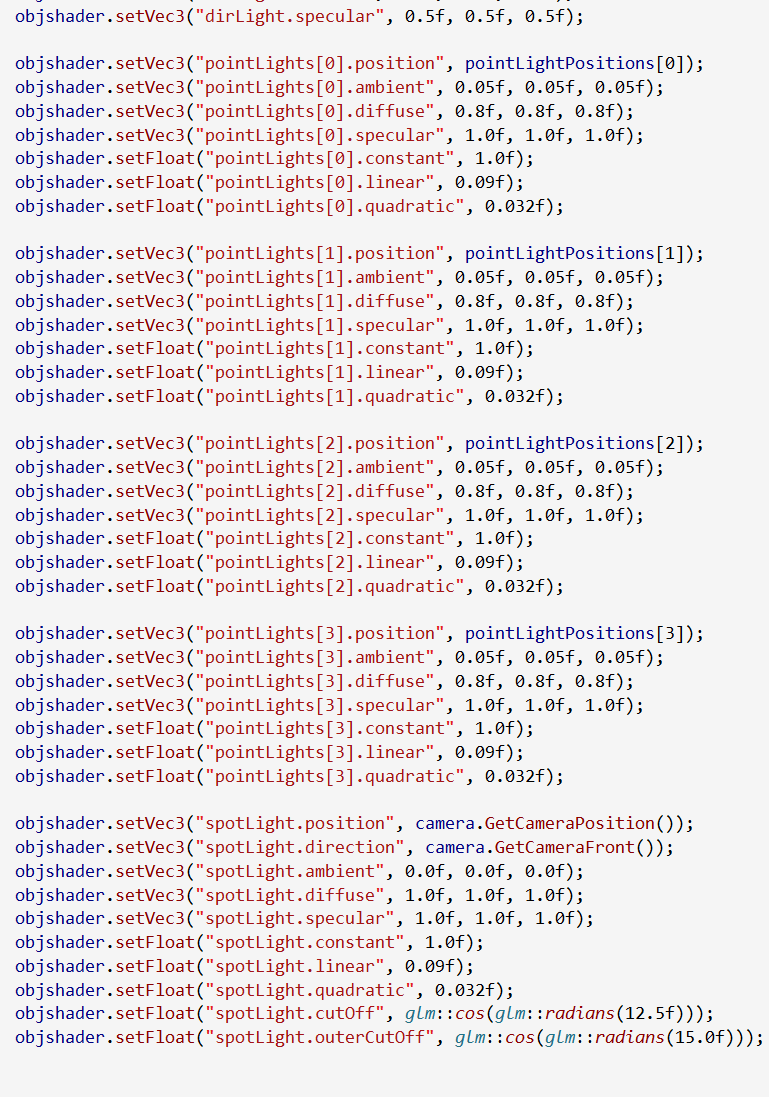


4.4 我们在场景中设置好多个点光源的位置，并按顺序绘制点光源模型。





4.5 按照三种投光物的光照属性，设置好对应的光照信息。



4.6 编译运行程序，我们即可得到多种投光物共同作用的光照场景。



4.7 我们还可以再玩点花的。通过调整背景颜色，以及三种投光物的颜色属性可创造出风格各异的渲染场景。

