**练 习 题 报 告**

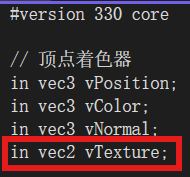
|  |
| --- |
| **课程名称 计算机图形学**  **项目名称 简单可扩展曲面纹理映射**  **学 院 计算机与软件**  **专 业 计算机科学与技术**  **指导教师 熊卫丹**  **报 告 人 杜良衡 学号 2022150255** |

1. **练习目的**
2. 了解三维曲面和纹理映基本知识
3. 了解从图片文件载入纹理数据基本步骤
4. 掌握三维曲面绘制过程中纹理坐标和几何坐标的使用

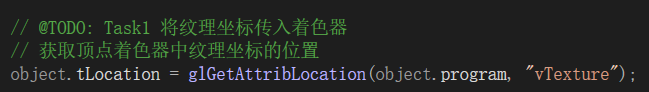
**二．练习完成过程及主要代码说明**

**Task1 将纹理坐标传入着色器**

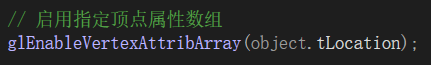
在顶点着色器内，设置了变量vTexture来接收纹理坐标并将其传递给片元着色器，因为纹理颜色的计算要在片元着色器内进行。



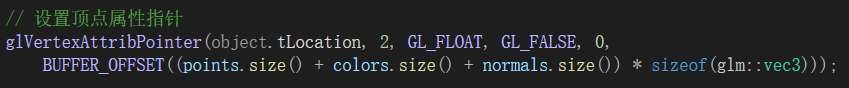
因此在MeshPainter.cpp的bingObjectAndData方法中，需要先从顶点着色器中获取纹理坐标，即vTexture变量的位置，通过调用glGetAttribLocation 函数来实现。



接着调用glEnableVertexAttribArray函数来启用指定的顶点数组，，用来存储纹理坐标数据。



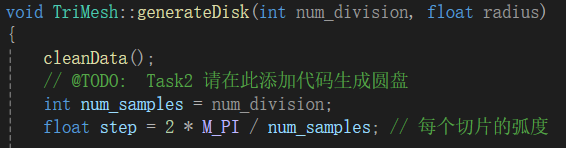
再利用glVertexAttribPointer函数设置纹理坐标属性指针，从而传递到着色器中。这个函数的参数分别为属性的位置、每个顶点的纹理坐标数量、纹理坐标的类型、是否需要归一化数据、顶点之间的字节数即步长以及偏移量。



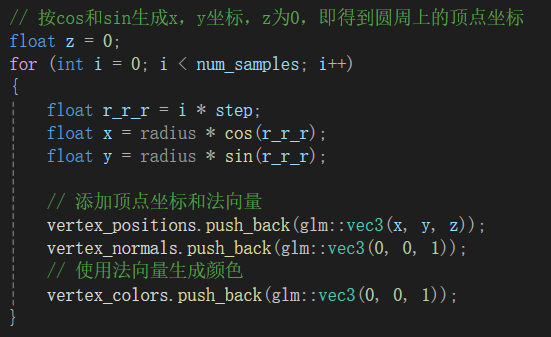
**Task2 生成圆盘与圆锥并贴图**

先在TriMesh.cpp中构建生成圆盘以及圆锥体的函数generateDisk和generateCone，可以参考已经构建好的生成圆柱体的函数generateCylinder。

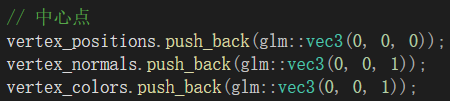
在生成圆盘的generateDisk函数中，先清空之前的模型数据，定义了num\_samples来表示圆周上顶点数量，它的值等于分段数num\_division，以及step来表示每个切片的弧度，即每个顶点之间的角度差。



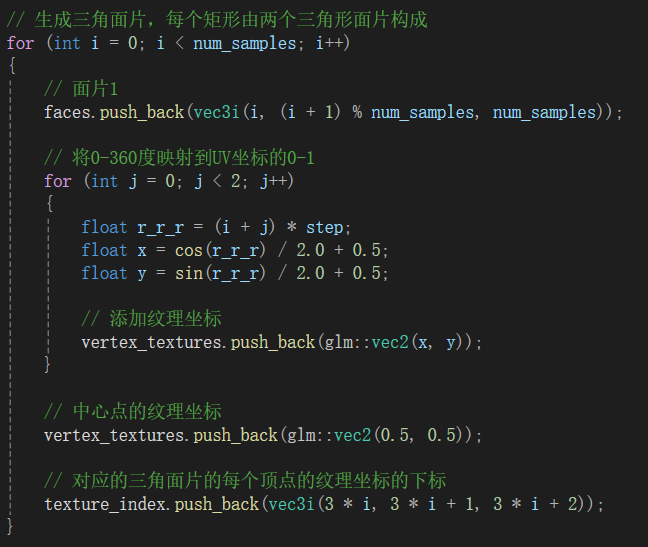
接着生成圆周上的顶点，通过cos和sin函数计算每个顶点的x和y坐标，z坐标固定为0。而添加的法向量设置为(0, 0, 1)，表示指向正Z轴，颜色设置为(0, 0, 1)，即蓝色。



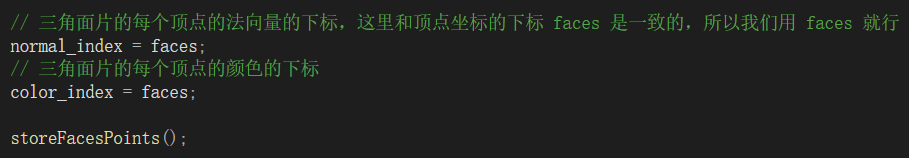
然后添加中心点的顶点坐标(0, 0, 0)、法向量和颜色，其法向量的颜色与圆周上的顶点相同。



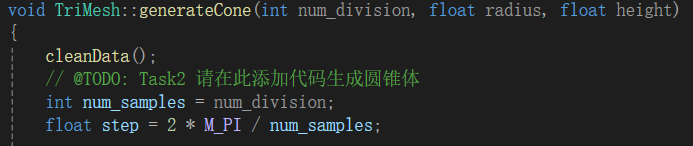
再通过循环生成三角面片，每个矩形由两个三角形面片构成，每个三角形面片由圆周上的两个顶点和中心点组成。在循环中，生成每个三角形面片的顶点索引，并映射UV坐标到三角形的每个顶点，从[0,2Π]映射到[0,1]。最后存储三角面片的每个顶点的纹理坐标的下标



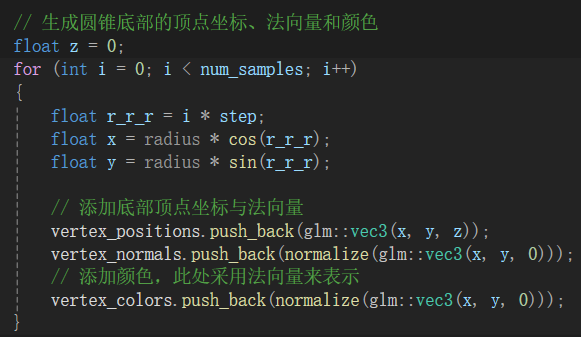
最后设置法向量和颜色的索引，与顶点索引一致，并调用storeFacesPoints方法存储面片信息。



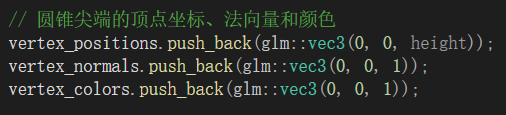
生成圆锥的函数generateCone的流程类似。先清理之前的数据，初始化num\_samples和step变量分别作为圆周上的顶点数量和每个切片的弧度。



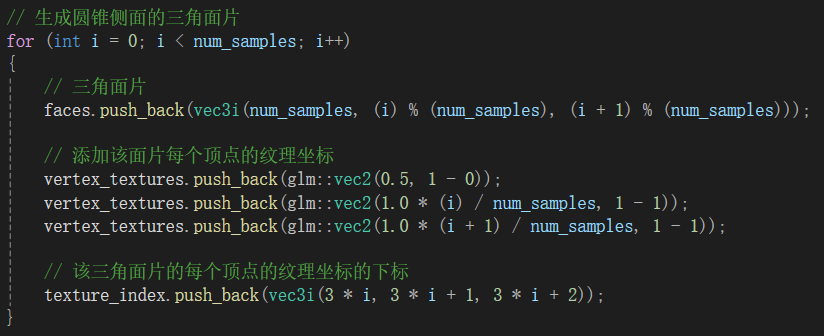
再通过循环生成圆锥底部的顶点。循环中通过cos和sin函数计算每个顶点的x和y坐标，z坐标固定为0。添加的法向量设置为(x, y, 0)的单位向量，指向圆周外侧。颜色使用法向量来表示，这样颜色会随着顶点的位置变化而变化。



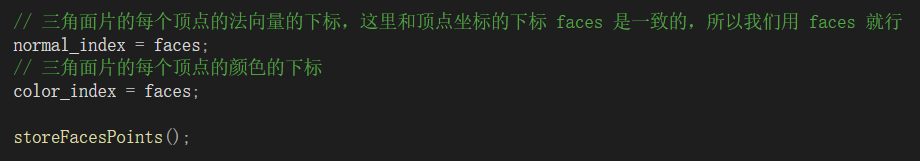
接着添加圆锥尖端的顶点、法向量和颜色。法向量设置为(0,0,1)，指向正Z轴。



然后通过循环来生成侧面的三角面片，每个三角面片由圆锥尖端和底部圆周上的两个顶点组成。将这些面片添加到面片数组faces中。添加该面片每个顶点的纹理坐标，需要从[0， 2Π]映射到 [0,1]，再将每个三角面片纹理坐标的下标存储在 texture\_index 中。

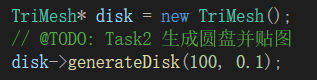


最后设置法向量和颜色的索引，与顶点索引一致，并调用storeFacesPoints方法存储面片信息。



在main.cpp的init方法中，仿照圆柱体的创建和贴图，来生成圆盘和圆锥对象并进行贴图。

以圆盘对象为例，先创建一个TriMesh 类的实例disk，并调用刚才构建好的generateDisk函数来生成圆盘。该方法接受两个参数，分别为分段数和半径。分段数用来控制几何体表面的平滑程度。



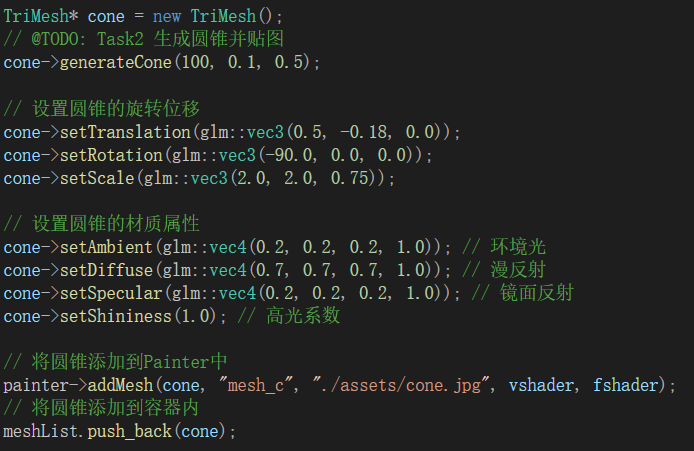
接着设置物体的旋转位移包括平移、旋转和缩放，再设置相应的材质属性，包括环境光、漫反射、镜面反射和高光系数。



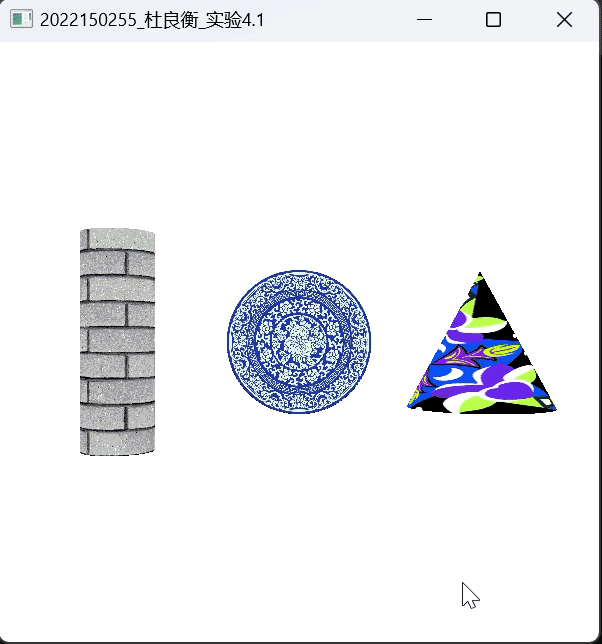
然后调用addMesh方法将圆盘对象添加到绘制器Painter中，并指定纹理与着色器。最后将圆盘对象添加到meshList容器中，该容器用来在程序结束时回收和删除创建的物体对象。



圆锥对象的生成和贴图过程与圆盘类似，不同的是在调用生成圆锥的函数generateCone中，需要传入三个参数，分别为分段数、半径以及高度，还有圆锥的旋转位移需要做对应的调整。



**最终效果展示**

****