**第一次实验报告**

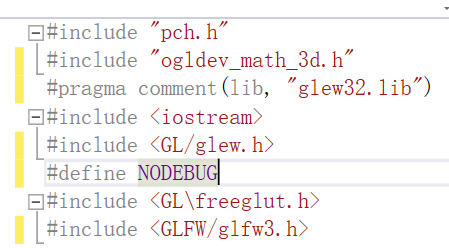
2019.3.3

19020182203646 姚添译

**实验目标：**利用OpenGL实现2维小球触地不断弹起的效果

**实验过程：**

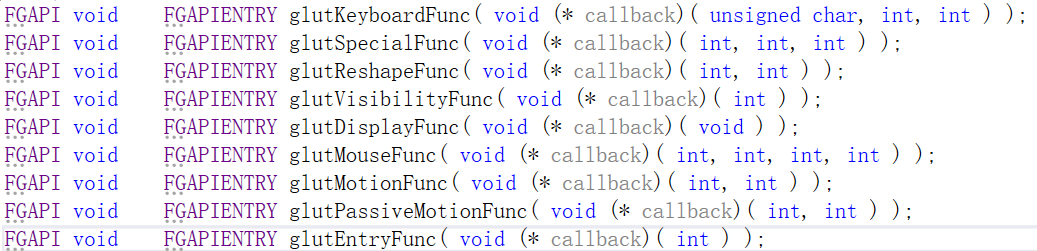
1. 准备工作
2. 搜集资料
   1. 在网上大致了解了OpenGL是图形硬件的软件接口，是一个具有较大可移植性的函数库，可以基于C语言运行。利用OpenGL可以高效率地进行图形的绘制。
   2. 大致翻了翻《OpenGL编程指南》和《基于OpenGL的自顶向下方法》两本书前面的内容，了解了一些基本的原理和具体程序的框架。
3. 环境安装

在实验课上机之前我自己先根据网上的教程安装了freeGLUT和glew库。由于自身对C语言项目经验过少，如何配置环境就成了一个问题。也是自己摸索了一番才搞明白导入库的方法，以及哪些环境变量需要手动配置。

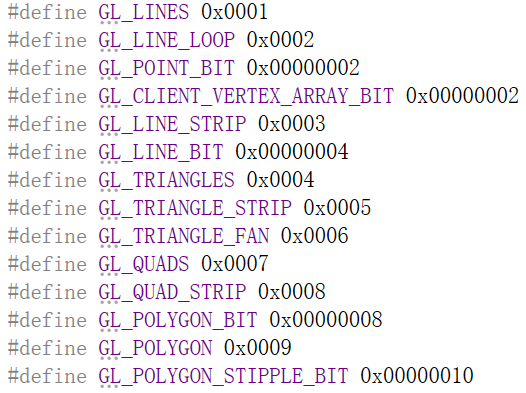
（加了几个可能用到的库。导入方式很奇葩，主要还是VS的问题……）

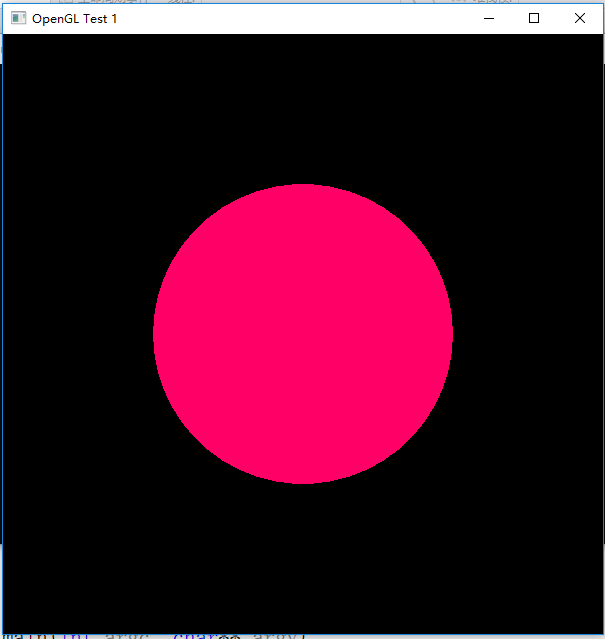
1. 分析以及编写程序
2. 写基本框架

基本框架包括main函数、display函数、init函数。OpenGL有一些事件回调函数需要注册特定的函数，比如显示回调函数应该把display作为参数输入，然后每当GLUT确定一个窗口的内容需要重新绘制时，回调函数就会运行，所以所有重绘场景的操作都应该放在display中。调用glutPostRedisplay() 会指示glutMainLoop()调用display进行重新绘制。glutMainLoop()用来启动程序，进入循环后不会停止。

演示代码中的glutReshapeFunc()，glutKeyboardFunc()分别对应当窗口形状改变时和接收到键盘输入时的事件回调函数。库里还定义了许多其他的事件回调函数，但是有些没法直观看出作用。

1. 绘制静态图像

OpenGL库主要是用来绘制3D图形的，所以绘制三维物体的函数比较多，这次只需要简单地画个圆，按理说会很简单。但是看了书、查了库之后却发现没有绘制圆的函数，只能画线、三角形、矩形、多边形：

无计可施的我决定近似。第一种，采用微分的方法应该是把圆分解成许多个扇形，再把扇形近似成三角形，但是这种方法绘制一个圆会把同一个函数调用过多次，性能会比较低。因此第二种方法可以用正n边形去近似圆，这样只需要计算一次n个顶点的位置，可以降低计算量。图为多边形顶点为50个的情况，可以看到已经近似得很好。

第三种方法就是直接画一个球，如果不加任何纹理和光照，在摄像机中就应该是一个圆。但是我对3D对象还没什么了解就暂且放弃了这种做法。

1. 动画

最初对OpenGL的动画根本不知道怎么实现。我以为会有类似cocos2d里面的物理引擎，可以直接设置一个对象（精灵）的速度属性，但是显然第一步绘制出来的图形已经离散化而不是一个整体的元素了，也就不能对其整体设置一个属性。后来看红宝书发现，动画的实现居然就是逐帧绘制静态图像……实在没想到OpenGL是采用这么底层的实现方法。仔细看了看书，书里介绍OpenGL采用双缓冲的做法，在一个缓冲区渲染，另一个缓冲区显示，因此只要重复绘制+交换缓冲区这个操作越快，人眼就越不会感觉掉帧。

那么如何让这个动作重复执行下去又成了问题。依照我之前写一些游戏小插件的经验，各种环境里应该都是有内置的时间回调函数的，但是我搜不到相关的函数。后来经过老师提醒，GLUT是内置了一个延时器的，就是：

递归调用这个函数就可以实现计时器的效果。至此动画部分的理论问题就扫清了。

1. 运动轨迹

小球抛出触底反弹最后停止的过程可以描述如下：

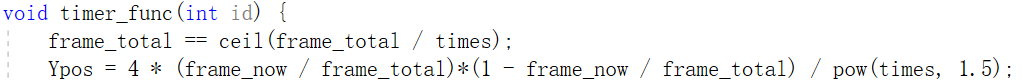


其中t是每周期帧数，是未定系数。

显然与碰撞的次数有关，代表着每次碰撞后能量的衰减，但我并不想每次靠解碰撞能量损失方程的方法去计算，而是想通过简单的函数对近似。

所以观察的性质：

1. 在第一次时要使，即第一次要弹到屏幕顶端；
2. 随次数增加而减小；
3. 次数越多，减小的速率越慢；
4. 快速收敛向0.

综上我最终选取作为近似的函数，并让次数每次碰撞后加1。通过测试，这个函数收敛的速度比较接近实际情况的能量衰减。

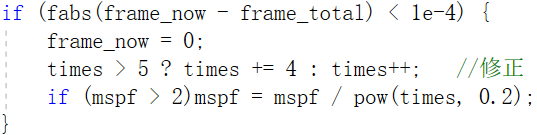
1. 刷新帧率

帧率(FPS)也是很重要的概念，这决定GLUT每秒应该绘制多少帧。如果我想让第一个碰撞周期持续秒，每次递增1，在一个周期里恰好达到的值，那么就有



那么重要的问题就是每个运动周期的确定。运动周期的长度显然与运动的次数有关，运动次数越多，运动周期越短，但是这种衰减也不能出现肉眼可见的违背物理现象的“越振越快”，于是还是可以采取近似的方式，大致取



再综合的取值对这个方程进行简单的修正：

就获得了比较好的效果。

**实验总结：**

这次作业让我大体熟悉了C语言中如何导入库文件，如何配置一个基本的工作环境，了解了OpenGL程序的框架和简单动画的实现方法，同时也对模拟物体运动的方法有了初步的认识。希望自己能够通过课程的学习不断增进对OpenGL的理解。