# 实验目的

学习并掌握基本的三维编程知识；学习如何使用三种方式生成基本的几何图元；学习设置视口的各个参数。

# 实验内容

1、分别以本章讲述的三种生成方式生成一个四面体：

① 直接使用针对单个顶点、单一数据源的API函数；

② 使用数组；

③ 使用顶点数组。

2、分别以如下的视口设置绘制犹他壶：

① glViewport( 0, 0, w/2, h/2 )

② glViewport( w/2, 0, w/2, h/2 )

③ glViewport( w/4, h/4, w/2, h/2 )

④ glViewport( 100, 50, 300, 300)

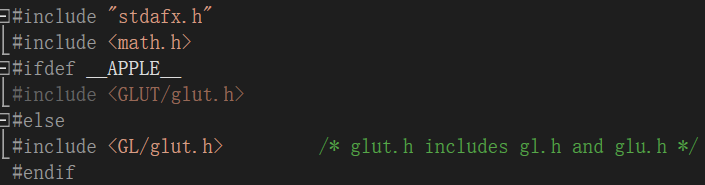
⑤ glViewport( w/2-W/2, h/2-H/2, W, H )，其中常数W<w, H<h。

根据结果，分析glViewport( )中各参数的几何意义？这些参数所在的窗口坐标系与GLUT中MouseCallback回调函数中参数所使用的窗口坐标系有什么区别？

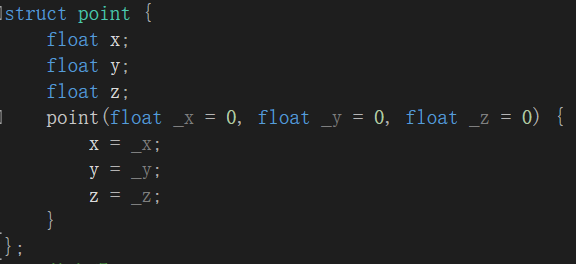
# 实验步骤

## 程序主体

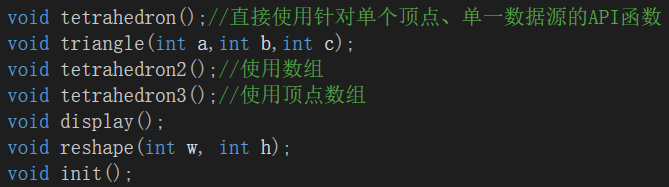
### 头文件



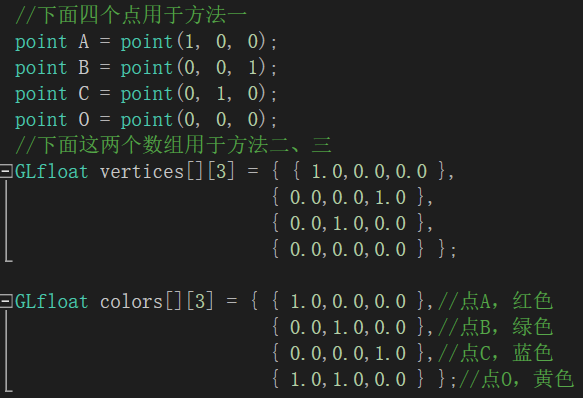
### 结构体定义



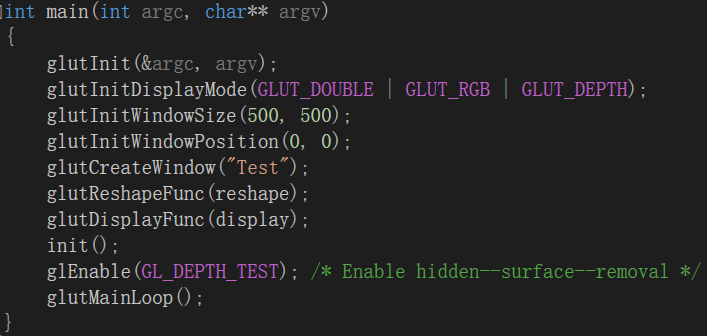
### 函数声明



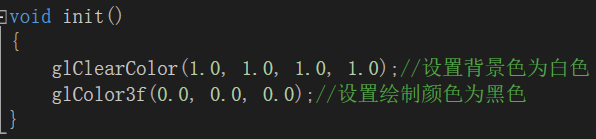
### 全局变量



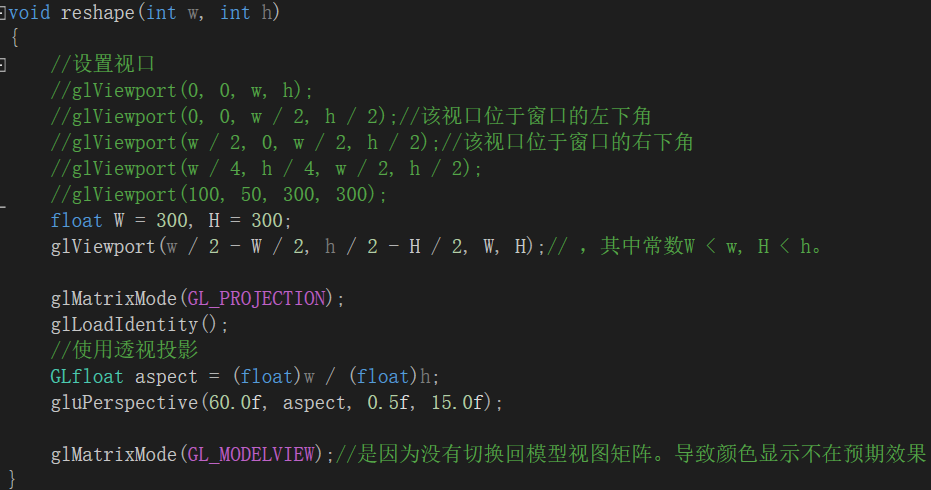
### main函数



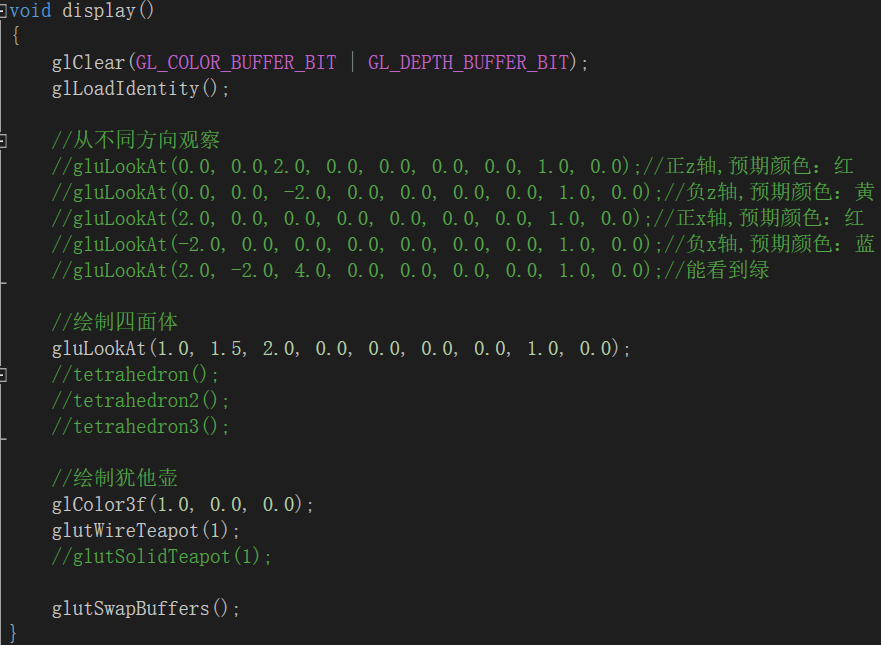
### init函数



### reshape函数

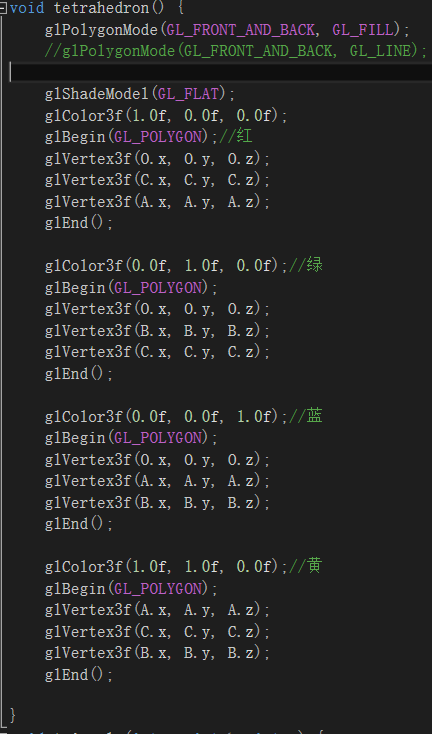


### display函数

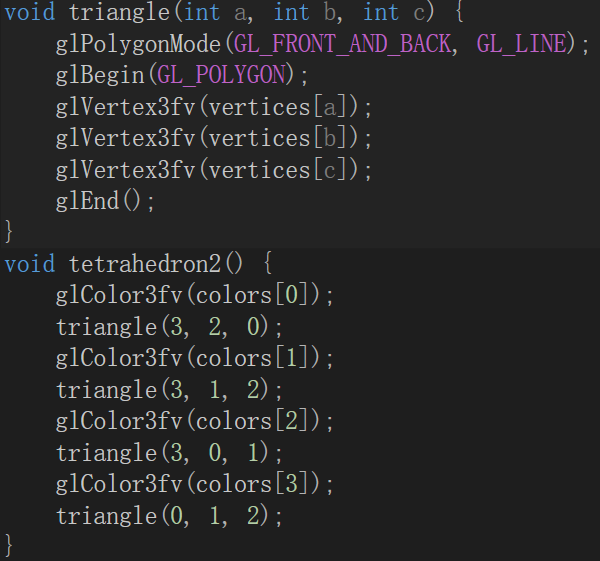


## 生成四面体

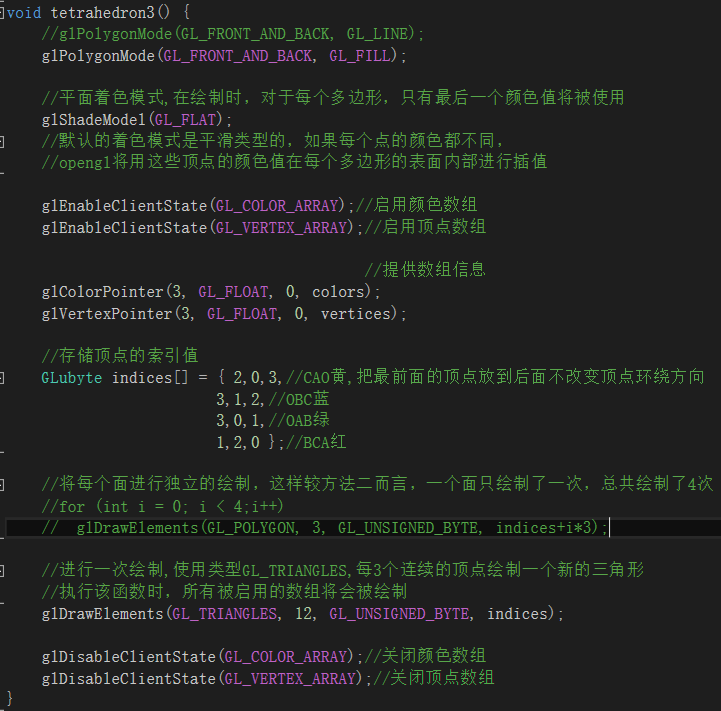
### 直接使用针对单个顶点、单一数据源的API函数



### 使用数组

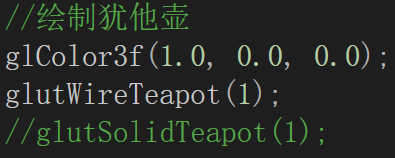


### 使用顶点数组

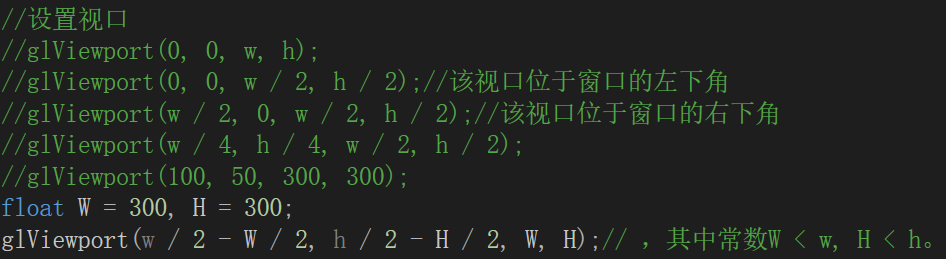


## 绘制犹他壶

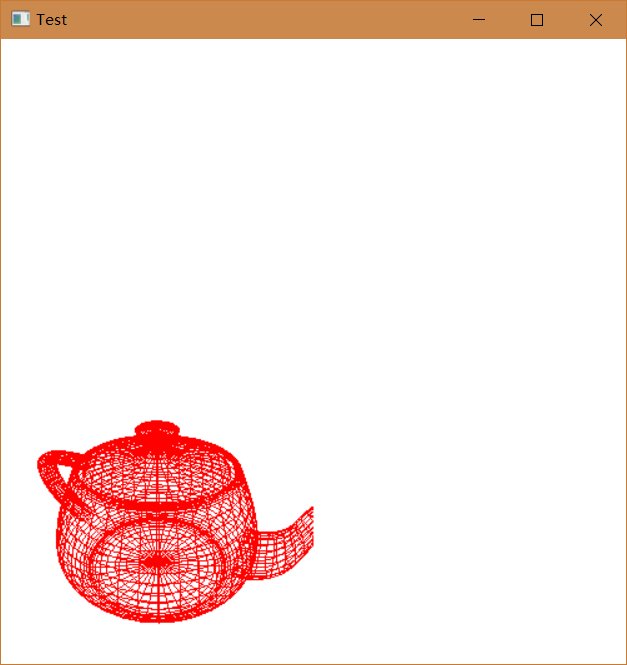
### display函数中：



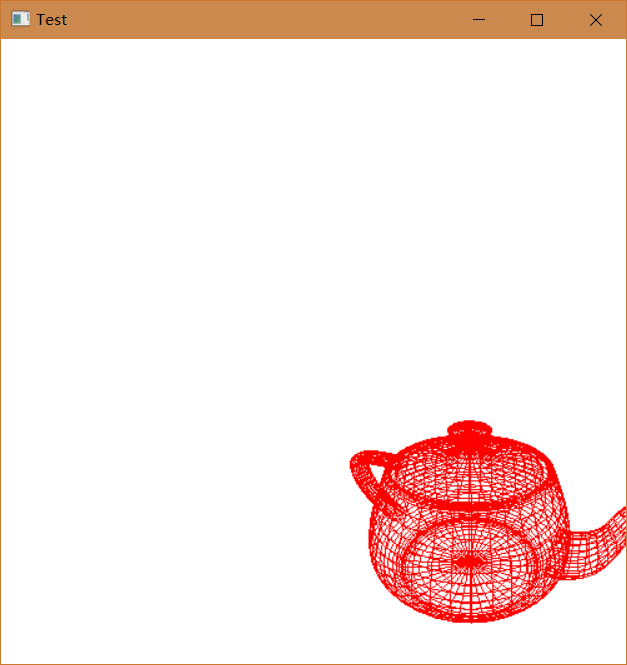
### Reshape函数中设置不同视口



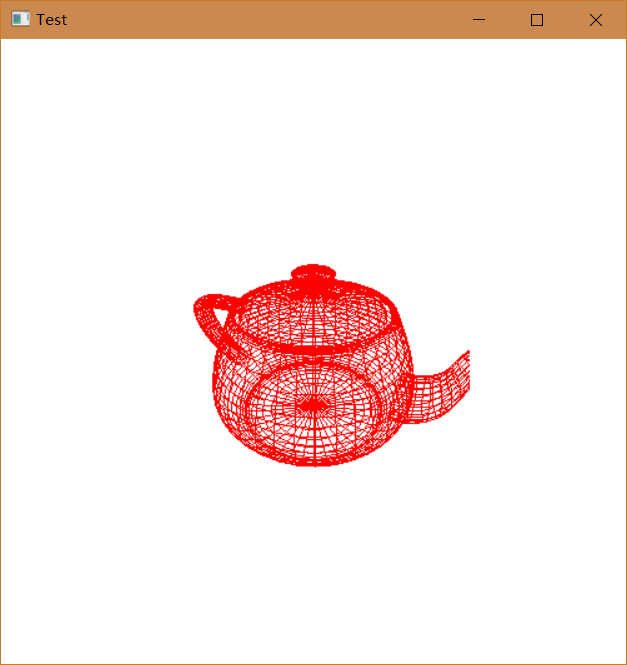
### 不同视口下的效果图



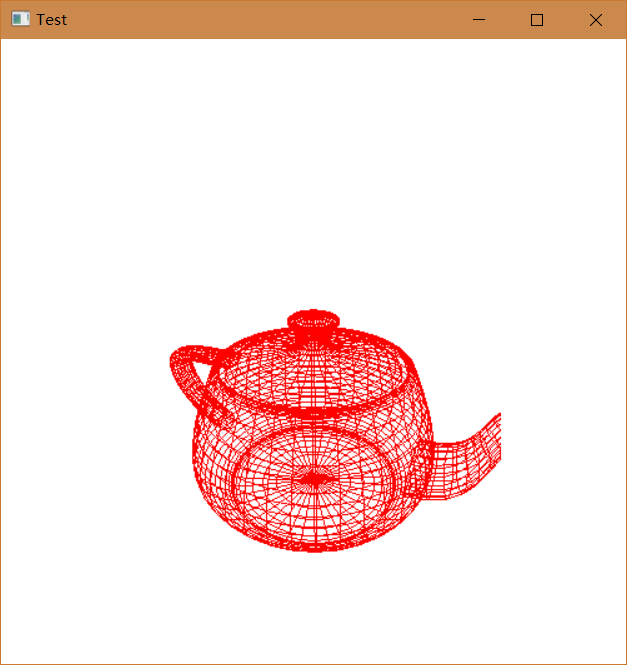
(0, 0, w / 2, h / 2)



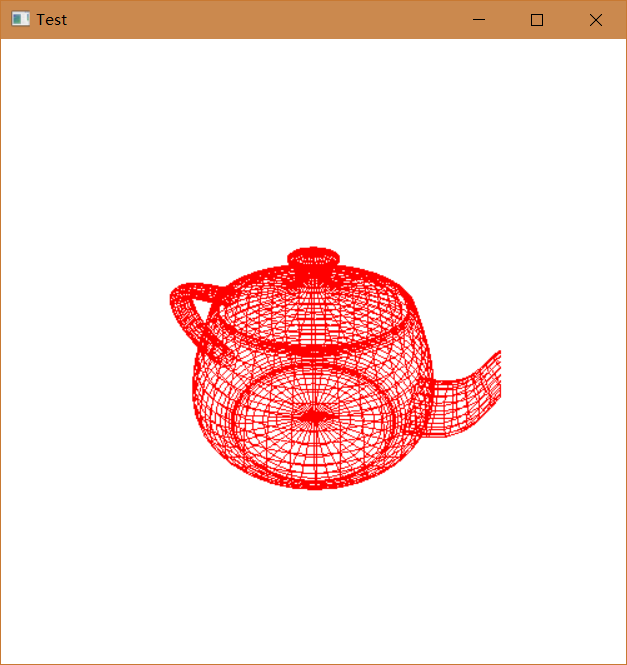
(w / 2, 0, w / 2, h / 2)



(w / 4, h / 4, w / 2, h / 2)



(100, 50, 300, 300)



float W = 300, H = 300;

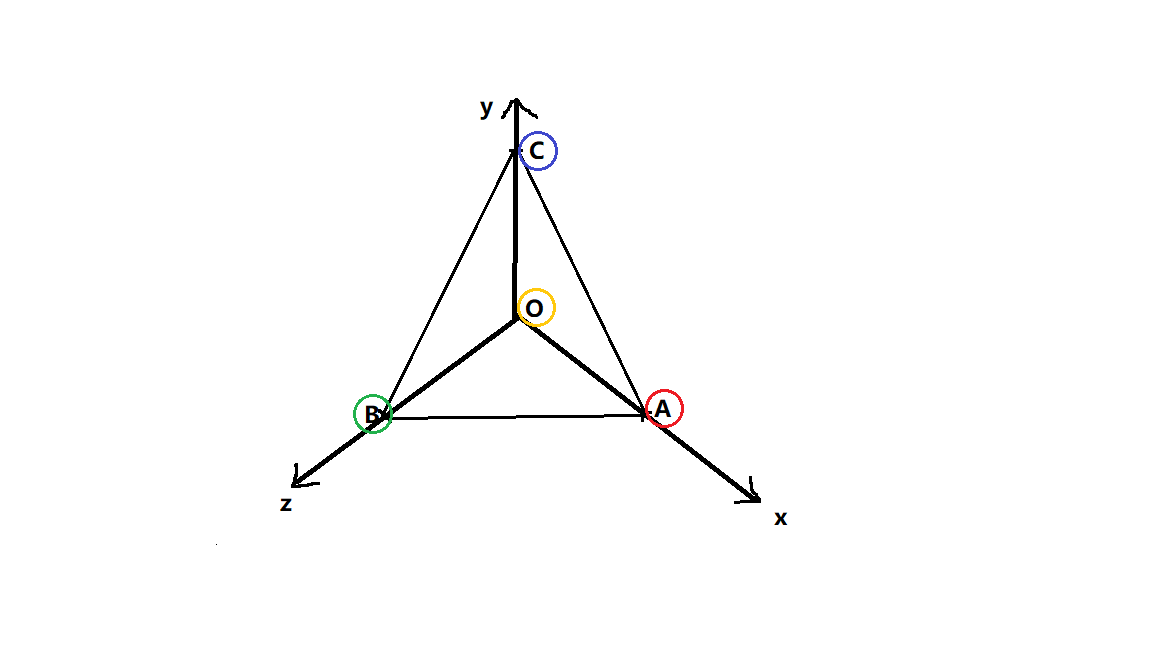
glViewport(w / 2 - W / 2, h / 2 - H / 2, W, H))

# 实验心得

## 生成四面体

将四面体绘制好后我从各个方向观察四面体，发现得不到预期的观察效果，颜色和预期的不一样。

后面还专门绘制了一张图来理清楚空间位置、颜色关系。



最后在老师的一番指点下终于找到出错的地方所在。

### reshape中：

glMatrixMode(GL\_MODELVIEW);//设置完透视投影后没有切换回模型视图矩阵

### display中：

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);//GL\_DEPTH\_BUFFER\_BIT没有加  
glLoadIdentity();//

## 绘制犹他壶

glViewport函数中，有四个参数，前两个参数是视口的原点坐标，后两个参数是视口的宽高。

视口的坐标原点是在左下角，而GLUT中MouseCallback回调函数中参数所使用的窗口坐标系的坐标原点在左上角。