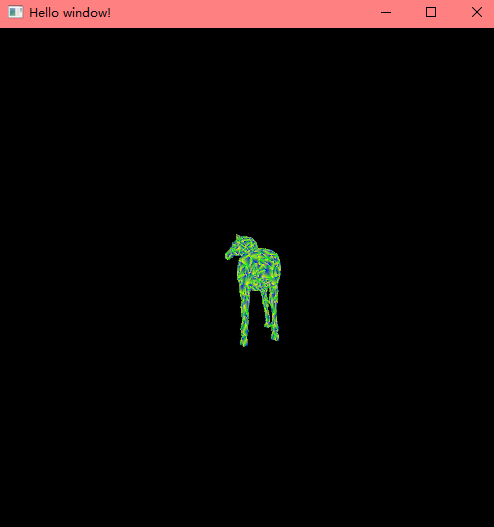
# 图形学实验报告02

## 实验要求：

* 模型显示：从给定的off文件(ftp//Models/cube.off、horse.off)读入网格模型的顶点位置和面，显示在屏幕上
* 观察：(1) 上下左右方向键移动模型；  
   (2) x键绕x轴方向旋转模型（模型中心为不动点）；  
   (3) y键绕y轴方向旋转模型（模型中心为不动点）；  
   (4) z键绕z轴方向旋转模型（模型中心为不动点）；  
   (5) s或S键缩小或放大模型（把模型拉远或拉近照相机）
* 撰写说明文档，描述你的程序的使用方法和运行效果。将说明文档和代码打包上传到ftp服务，压缩包的名称为“你的学号\_姓名\_上机作业\_2”。

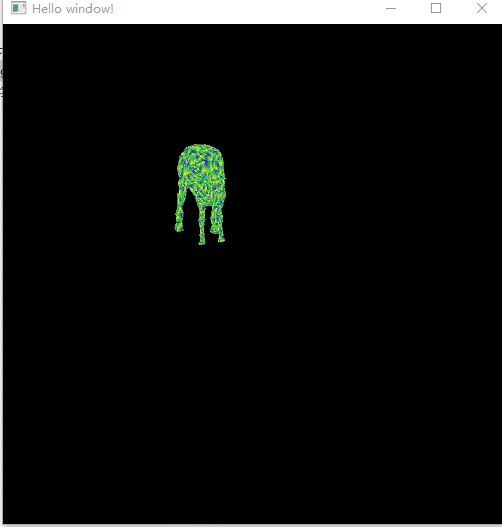
## 说明文档

原始图像：



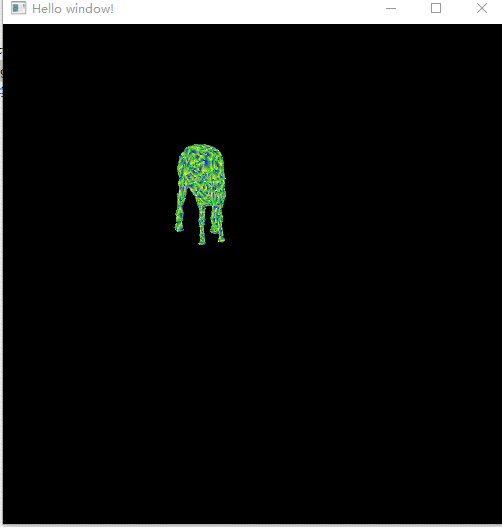
* 上下左右移动：文件附带的move.gif

使用： 进行操作



* 以模型中心为中心绕轴旋转：文件附带的rotate.gif

使用进行操作



* 放大缩小：文件附带的scalf.gif

使用 进行操作



## 实现代码

### 数据读取和存储

使用提供的读取函数进行读取，自定义存储结构：

// 存储点位置，缩放大小

typedef std::array<double, 3> point\_posi\_t, scalef\_t;

// 存储一组点位置 用于存储模型的坐标信息

typedef std::vector<point\_posi\_t> positions\_t;

// 存储一个面的位置索引

typedef std::vector<int> surface\_t;

// 存储整个模型，由一组面组成

typedef std::vector<surface\_t> surfaces\_t;

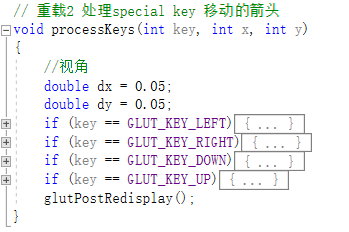
### 移动

* 存储模型中心位置：

//模型中心的世界坐标

point\_posi\_t model\_posi;

* 注册对specialKey的监听，处理函数：



### 旋转和缩放

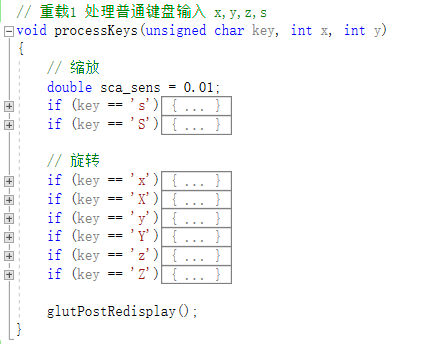
* 旋转和缩放信息存储：

scalef\_t scalf = {1.0, 1.0, 1.0};

//旋转角度

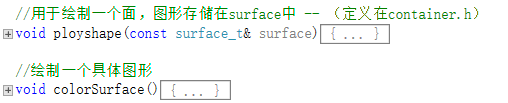
static GLfloat theta[] = { 0.0,0.0,0.0 };

* 注册对普通按键的监听，处理函数：



### 图像显示

* 在display中调用整个模型绘制的函数，模型绘制函数调用另一个绘制面的函数对模型的所有面绘制：



* display和变换：
  + 中间一部分为对模型进行的变换
  + 红线为模型绘制的函数

