

# 练习题报告

课程名称 计算机图形学

项目名称 OpenGL 鼠标键盘的交互

学 院 计算机与软件学院

专 业 软件工程（腾班）

指导教师 熊卫丹

报 告 人 洪子敬 学号 2022155033

## 一、练习目的

1. 学会 OpenGL 的简单键盘交互。
2. 掌握 OpenGL 的简单鼠标交互。
3. 进一步巩固 OpenGL 的基本图元绘制基础。

## 二. 练习完成过程及主要代码说明

练习题目：在主窗口中添加响应鼠标滚轮滚动的函数 `void sroll_callback(GLFWwindow* window, double xoffset, double yoffset)`，要求实现的功能为通过鼠标滚轮的上（下）滚动控制窗口中图形的顺（逆）时针旋转，并在 `main()`函数中使用 `glfwSetScrollCallback()`函数进行链接绑定。

解答：首先编写 `sroll_callback` 函数，我们观察题意和所给代码，发现我们只需要计算图形偏移的角度 `offsetAngle`，而偏移角度由偏移量和每一偏移量对应的旋转角度决定；而不难发现，我们滚动鼠标只有垂直方向会有偏移，即 `y` 方向上的 `yoffset`。因此我们可以得到旋转角度的计算公式：

$$\text{offsetAngle} -= \text{yoffset} * \text{delta}$$

至于为什么这里是减而不是加的原因如下：

默认的旋转正方向是逆时针，此时如果向上滚动鼠标，产生的 `yoffset` 是正值，即旋转的角度为正值，我们想要它是顺时针，所以需要它是负值；反之，向下滚动鼠标，产生的 `yoffset` 是负值，即旋转角度为负值，我们想要它是逆时针，所以需要它是正值。因此，我们直接减去它即可达到相反的效果，实现我们的目的。代码如下：

```
void sroll_callback(GLFWwindow* window, double xoffset, double yoffset) {  
    //向上滚动为顺时针旋转，向下滚动为逆时针  
    //鼠标滚动只有y方向发生偏移，用偏移量乘以角度变化量得到最终的角度变化，由于要求和实际相反，所以要减去而不是加上  
    offsetAngle -= yoffset * delta;  
}
```

HZJ

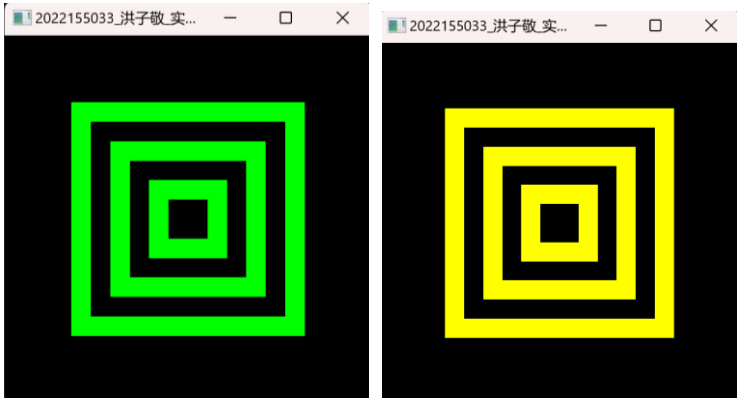
为了达到最终效果，我们还需要在 `main` 函数里面绑定鼠标滚轮回调函数，代码如下：

```
glfwMakeContextCurrent(mainwindow);  
glfwSetFramebufferSizeCallback(mainwindow, framebuffer_size_callback);  
glfwSetKeyCallback(mainwindow, key_callback);  
glfwSetMouseButtonCallback(mainwindow, mouse_button_callback);  
// @TODO: 创建鼠标滚轮回调函数并绑定。  
glfwSetScrollCallback(mainwindow, sroll_callback);
```

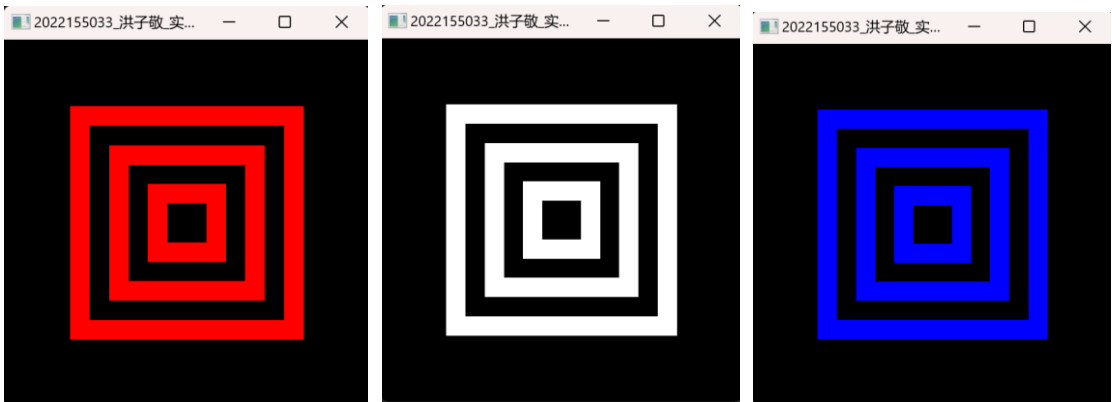
HZJ

代码的运行结果展示如下：

鼠标左键（左图）和右键（右图）效果如下：



键盘“R”键（左图）、“W”键（中间图）和“B”键（右图）效果如下：



鼠标滚轮向上滚动（左图）和向下滚动（右图）示例效果如下所示：

