

计算机图形学实验

实验6、RayTracing

姓 名：\_潘腾凯\_\_\_\_\_\_\_\_\_\_\_\_

学 号：\_37220232203786\_\_\_\_

学 院：\_\_信息学院\_\_\_\_\_\_\_\_\_

专 业：\_\_软件工程\_\_\_\_\_\_\_\_\_

年 级：\_2023\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2025年 5月 22 日

目录

[Task1.6：正确设置光线 3](#_Toc32445)

[1.（情况一） 3](#_Toc21797)

[Task1.7：实现阴影 3](#_Toc11650)

[1.（情况一） 3](#_Toc29667)

[Task1.8：实现镜面反射 5](#_Toc19401)

[1.（情况一） 5](#_Toc13995)

[Task1.9：实现折射 6](#_Toc824)

[1.（情况一） 6](#_Toc1832)

[（附加）Task1.10： 7](#_Toc19199)

[1.（情况一） 7](#_Toc11062)

# Task1.6：正确设置光线

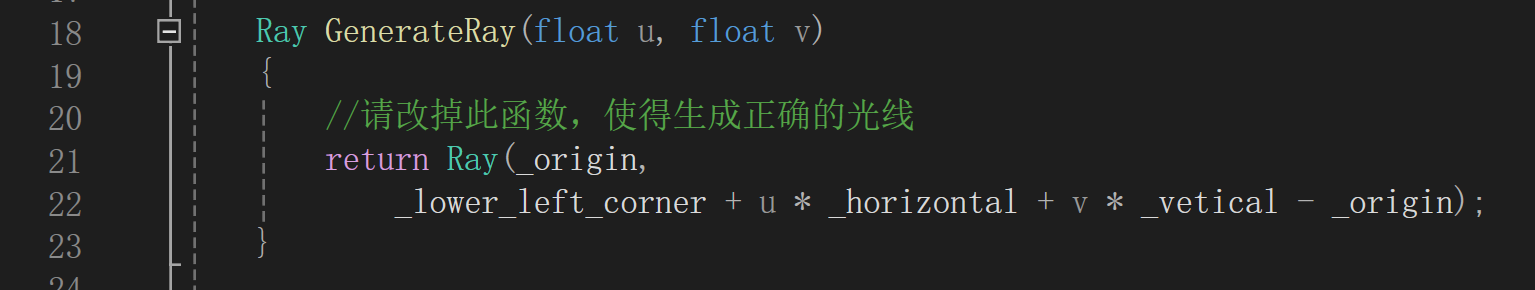
## 1.（情况一）

操作流程：

修改camera.hpp中生成光线的函数

Ray GenerateRay(float u, float v)

关键代码截图：



运行结果截图：

# Task1.7：实现阴影

## 1.（情况一）

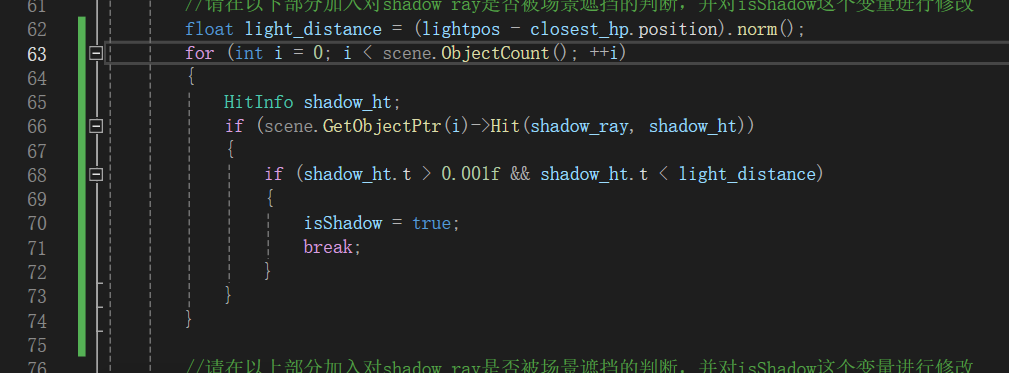
操作流程：

在RayTracer.hpp的函数

Vector3f RayColor(const Ray& ray, Scene& scene, int depth=0, bool test=false)

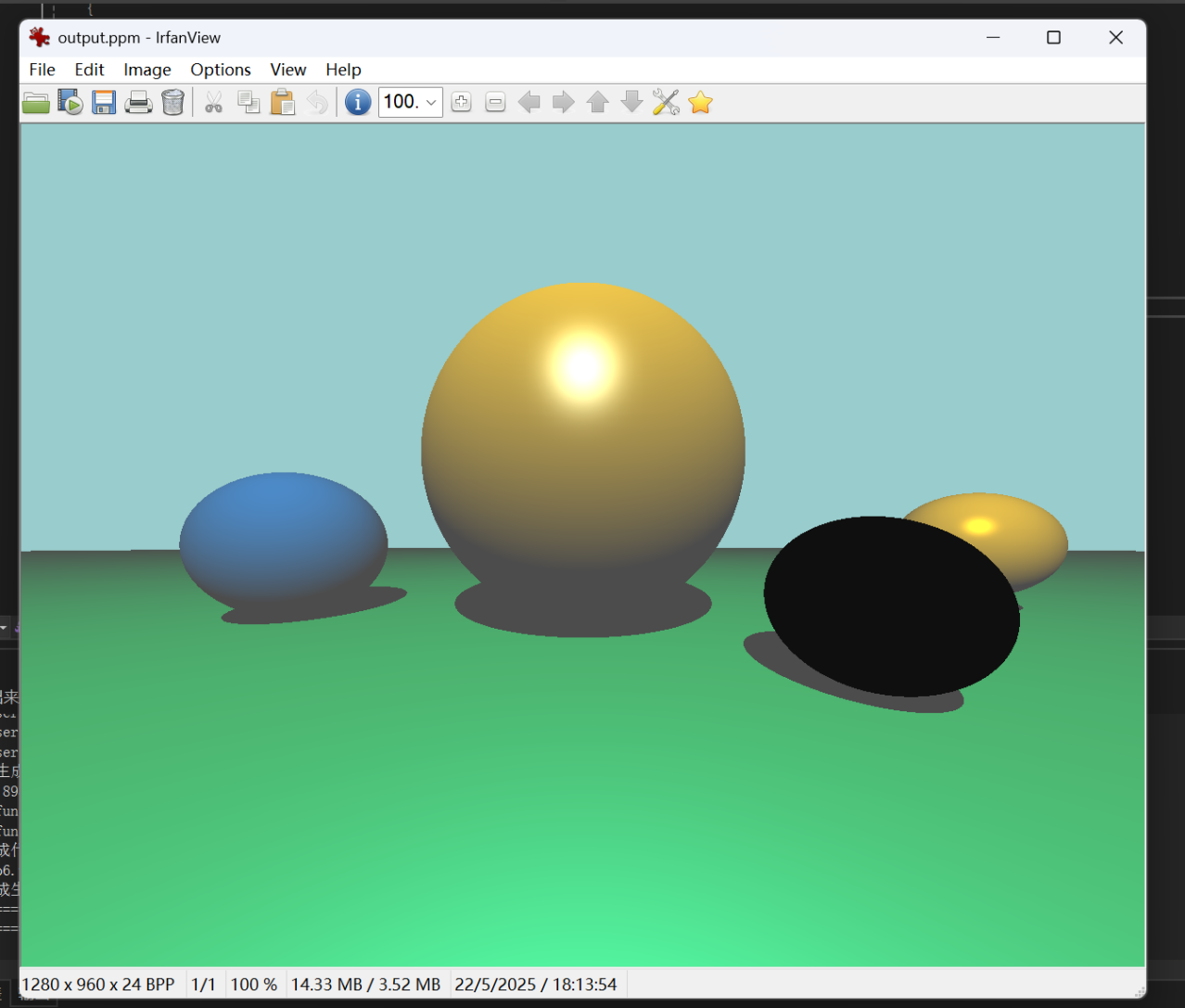
加入shadow ray是否能看到光源的代码，以生成阴影。

关键代码截图：



运行结果截图：

小球都产生阴影



# Task1.8：实现镜面反射

## 1.（情况一）

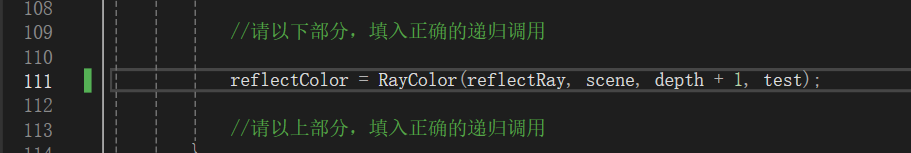
操作流程：

在RayTracer.hpp的函数

Vector3f RayColor(const Ray& ray, Scene& scene, int depth=0, bool test=false)

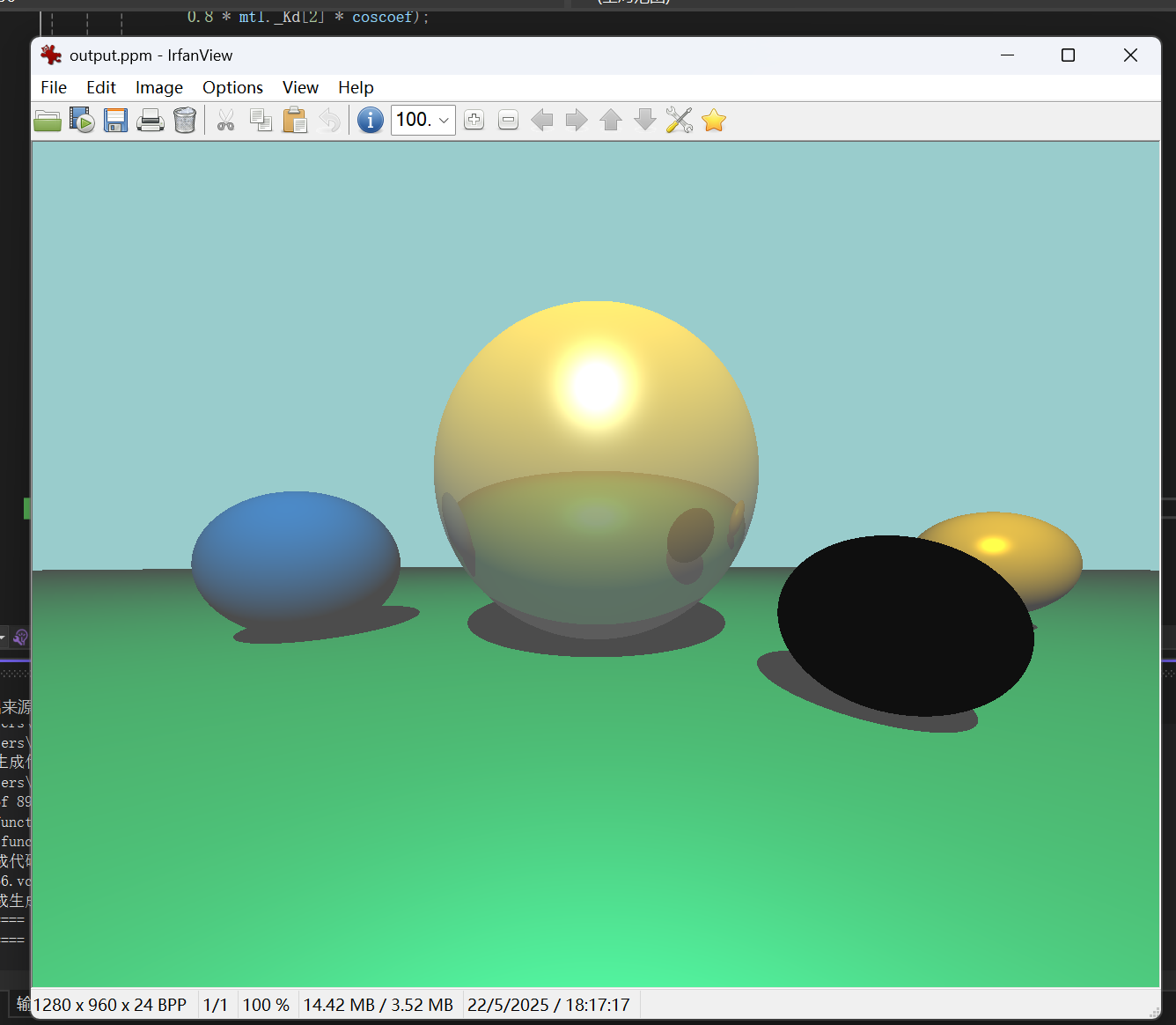
加入完成镜面反射的递归代码。

关键代码截图：



运行结果截图：

中间黄色圆球能观察到镜面反射效果。



# Task1.9：实现折射

## 1.（情况一）

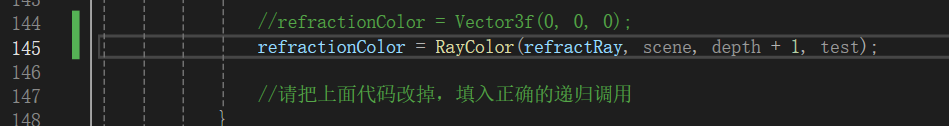
操作流程：

在RayTracer.hpp的函数

Vector3f RayColor(const Ray& ray, Scene& scene, int depth=0, bool test=false)

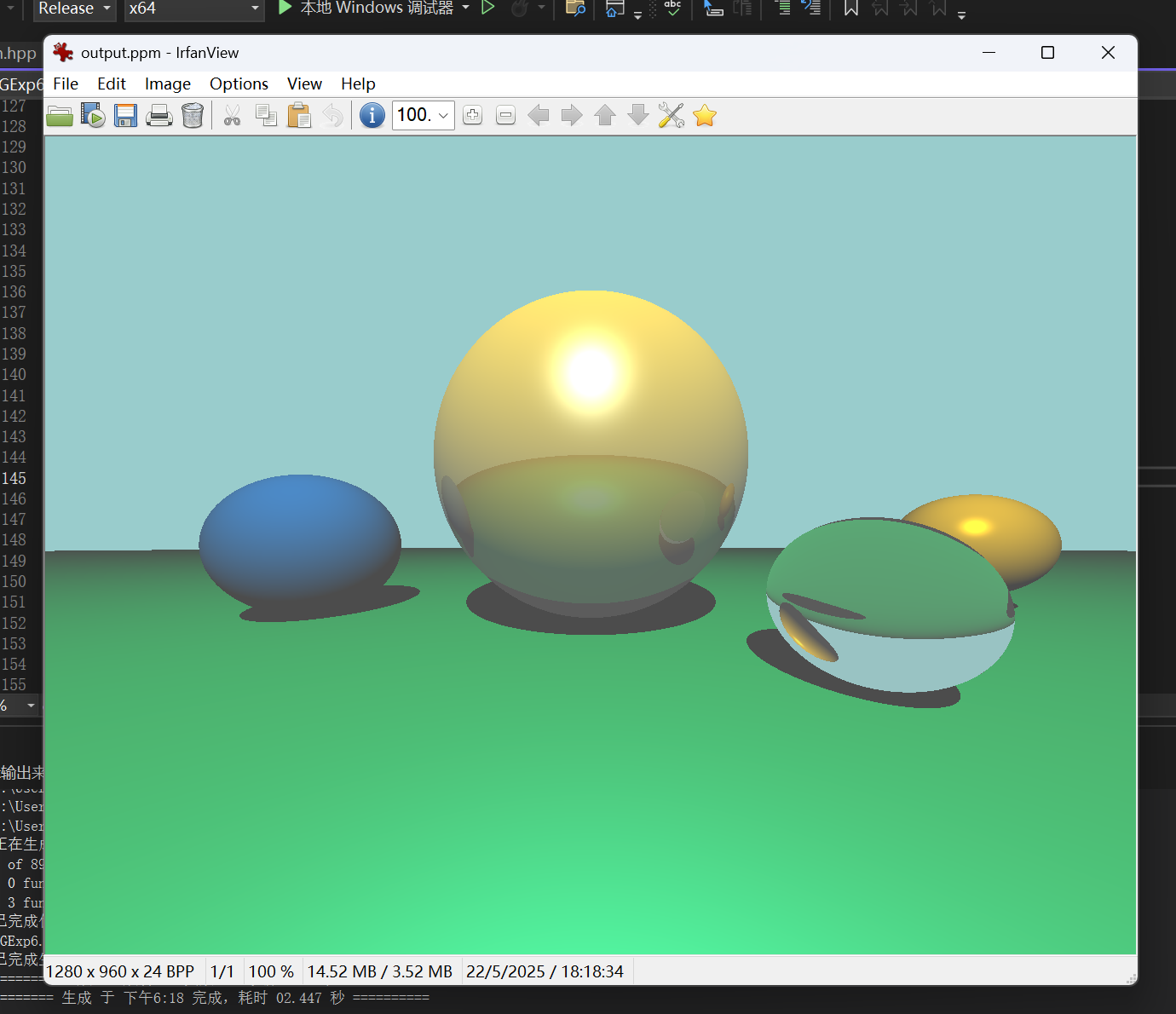
加入完成折射的递归代码

关键代码截图：



运行结果截图：

右侧的椭圆小球能够折射光线。



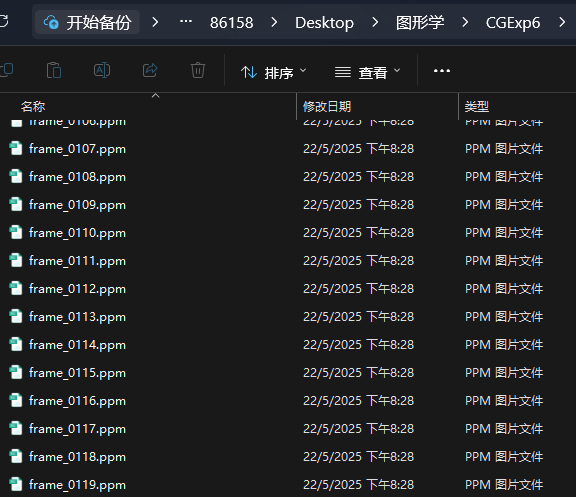
# （附加）Task1.10：

## 1.（情况一）

操作流程：

修改exp6.cpp（见task1.10.cpp）

输出120帧ppm文件，按顺序命名。



最后合成gif。（说明：因为Irfanview找不到生成gif的功能，网上找的版本也没有，就改用magick将ppm文件转为gif，操作过程如下）

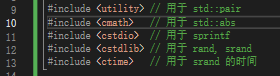


将该目录下所有ppm文件以1/100秒的间隔合成gif。

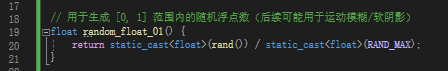
关键代码截图：

（完整见目录下task1.10.cpp）

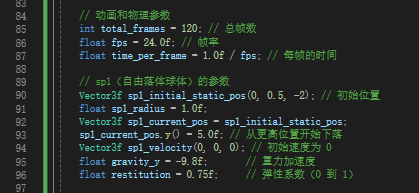
引入一些必要的库



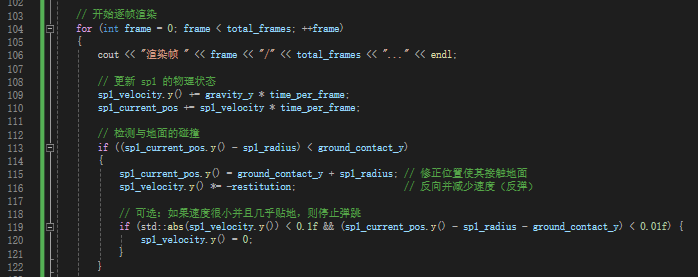
随机数生成：



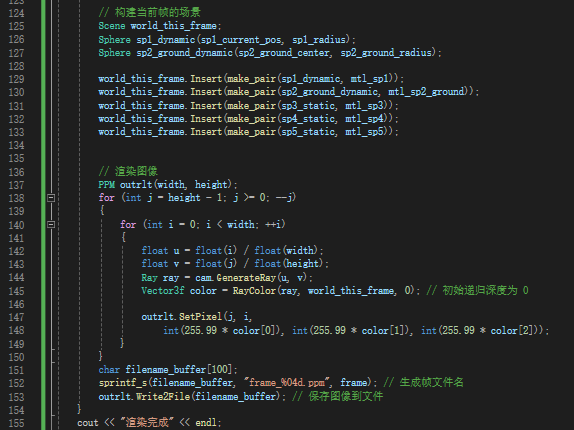
Gif参数设置：



模拟物理情况：



构建各帧：



运行结果截图：

