

## Planet.cpp

构建了一个行星类，其成员函数包括：

`calculatePosition(float time)` 根据时间计算行星所在的位置（行星在 **xy** 平面上）。

`calculatePosition(float time, float* vec)` 根据时间和一个平面的法向量计算行星所在的位置（可以在任意平面上）

`render(void)` 绘制行星，根据 `calculatePosition` 函数计算的行星位置通过相关的矩阵变换绘制行星。

`renderOrbit(void)` 绘制行星轨道（**xy** 平面上）

`renderOrbit(float* vec)` 绘制行星轨道（根据给定的法向量）

`getPosition(float* vec)` 获取行星的位置

`getRadius(void)` 获取行星的半径

`addMoon(float distanceFromPlanet, float orbitTime, float rotationTime, float radius, GLuint textureHandle)` 为指定行星添加卫星，输入参数包括卫星和行星的距离，公转周期，自转周期，半径，以及纹理句柄。

## Moon.cpp

构建一个卫星类，其成员函数与行星类类似，再次不赘述。

## Solarsystem.cpp

构建一个太阳系类，其成员函数用于调用相关的行星类和卫星类函数，绘制行星和卫星。

## Tga.cpp

用于生成纹理。

## Camera.cpp。

生成一个照相机类，其成员函数包括：

`setDefault(void)`：设置初始情况下照相机位置和方向

`transformOrientation(void)` 调用 `gluLookAt` 函数设置照相机

`pointAt(float* targetVec)` 照相机对准指定位置

`speedUp(void)` 照相机移动速率加快

`slowDown(void)` 照相机移动速率减弱

`forward(void)` 照相机向前移动

`backward(void)` 照相机向后移动

`left(void)` 照相机向左移动

`right(void)` 照相机向右移动

`yawLeft(void)` 照相机向左偏航

`yawRight(void)` 照相机向右偏航

## Main.cpp

利用 `SolarSystem` 类的成员函数生成行星和卫星，通过 `TGA` 类导入 `tga` 文件为行星添加纹理。同时支持根据相应的键盘操作调用 `camera` 类中的成员函数调整照相机的方向。

本程序最开始绘制了八大行星（在同一轨道平面上）、冥王星（与八大行星轨道平面有夹角）、地球的卫星月球。

键盘使用说明：

<和>用于加快和减慢照相机移动速度；

-和=用于加快和减慢时间流逝速度；

WASD 实现移动前后左右移动照相机；

QE 实现照相机向左向右偏航;

1-9 用于使照相机对准指定的行星, 在对准指定行星的时候, 可以通过 M 增加卫星;

[和]用于增大和减小行星的大小;

r 用于使照相机返回初始状态;

o 用于显示或隐藏星球的轨道;

1-水星 2-金星 3-地球 4-火星 5-木星 6-土星 7-天王星 8-海王星 9-冥王星