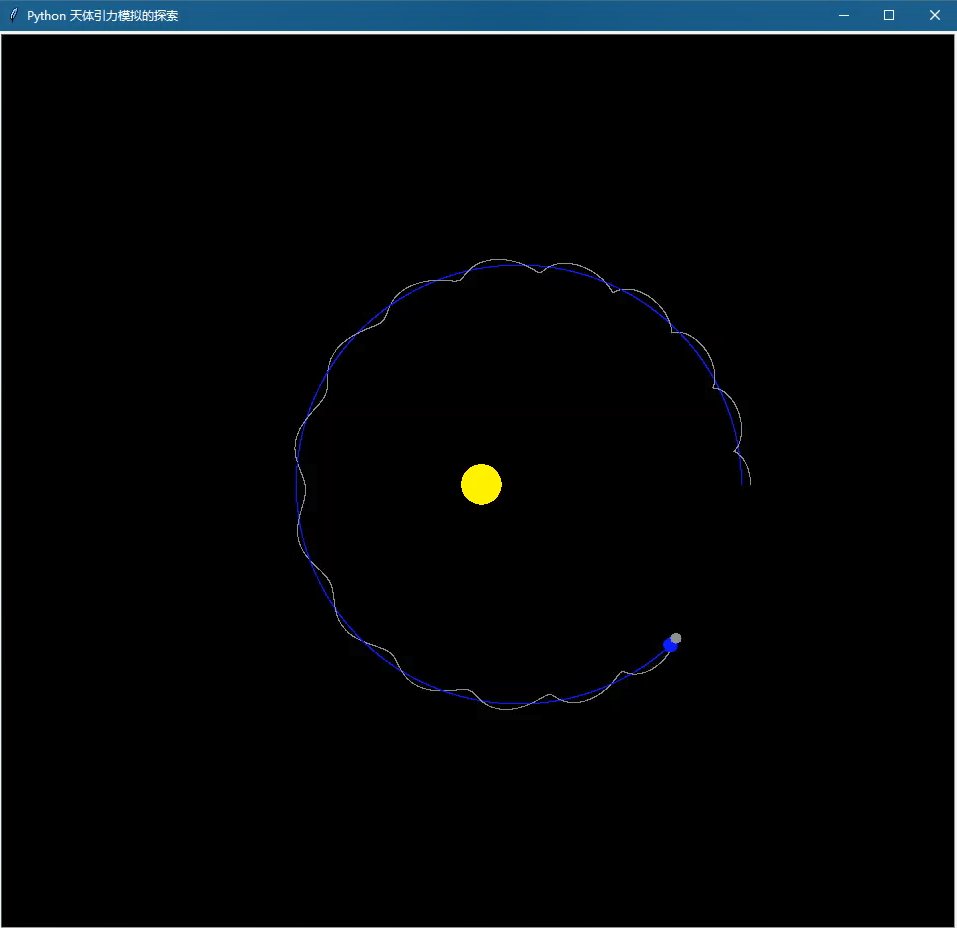
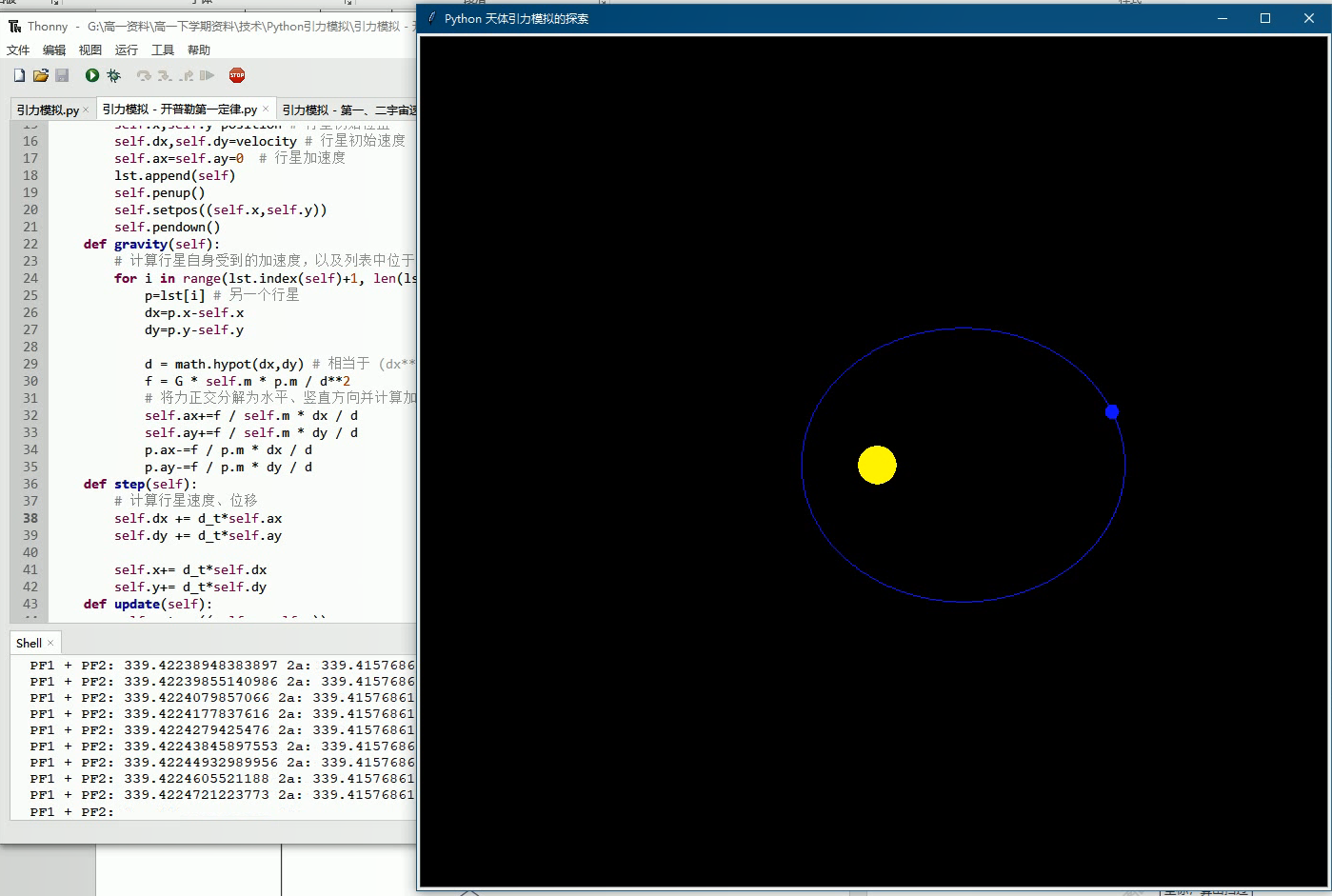
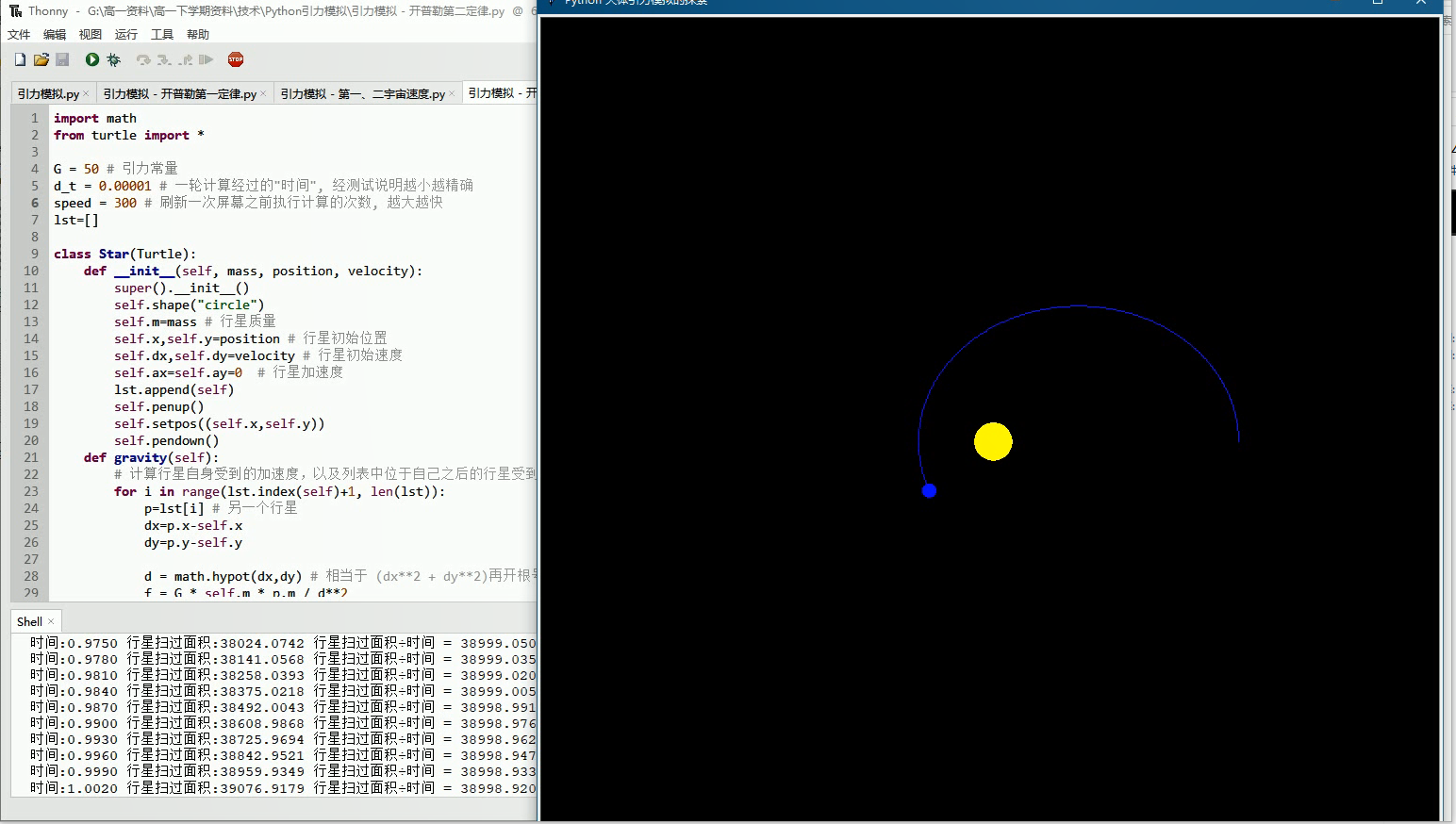
#### 作品效果图



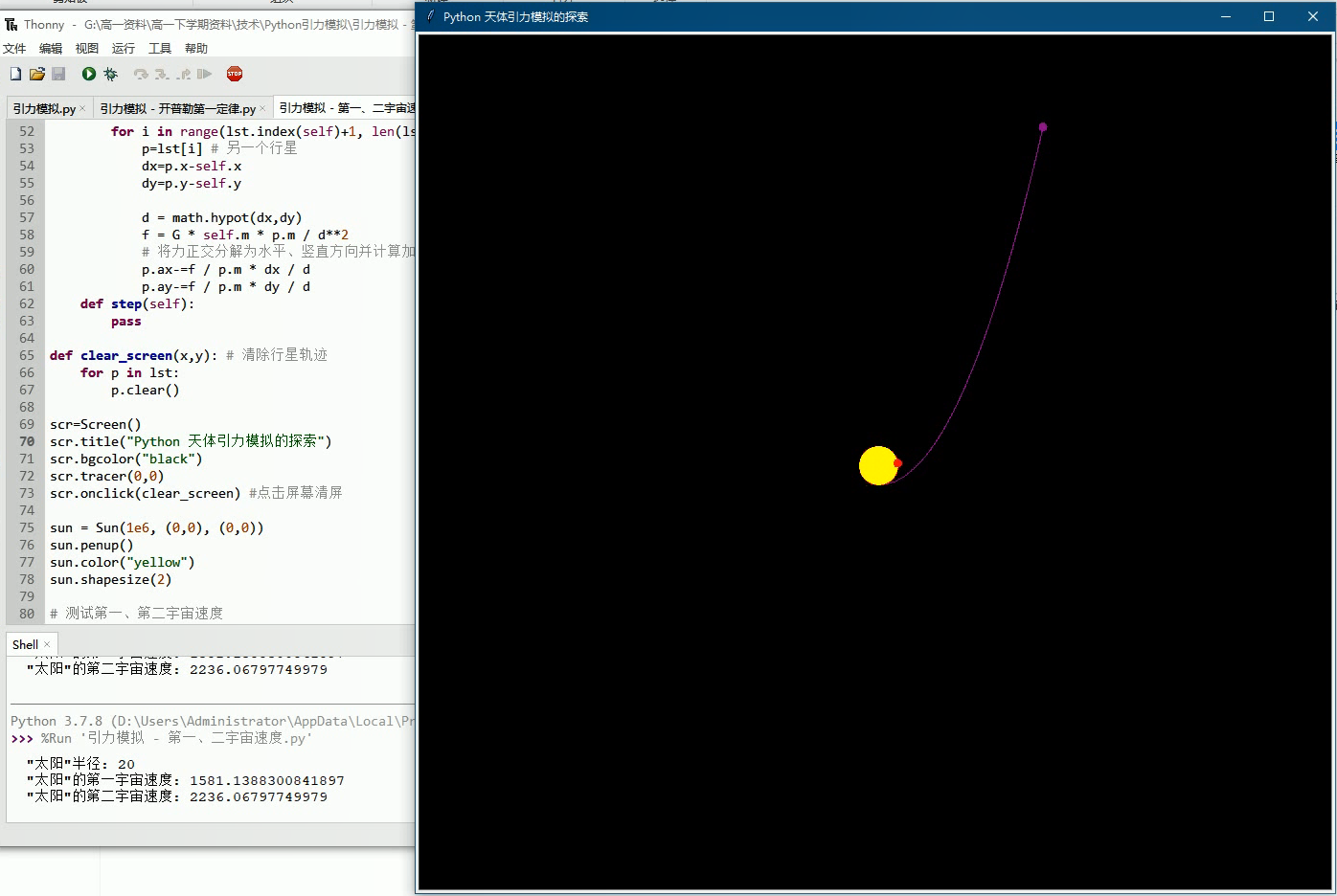
主程序运行截图



验证开普勒第一定律截图



验证开普勒第二定律截图



验证第一、二宇宙速度截图

#### 二、作品主题:

作者学习了高中物理《必修二》后，自己虽然明白了万有引力定律与第一宇宙速度、天体圆轨道之间的关系。但万有引力定律与第二宇宙速度，以及与开普勒第一、第二定律的关系在物理书中没有直接的推理过程。为此，编写了本程序, 研究万有引力定律与天体轨道行为之间的关系，以及验证开普勒第一、第二定律。

作品Python天体引力模拟的探索，在物理课本的公式基础上加以创新而成，可以模拟出天体的椭圆轨道、以及太阳系中的太阳、行星及卫星。

作品中，中心的黄色天体为中心的恒星，蓝色天体为行星，灰色天体为行星的卫星。天体具有运行轨道。

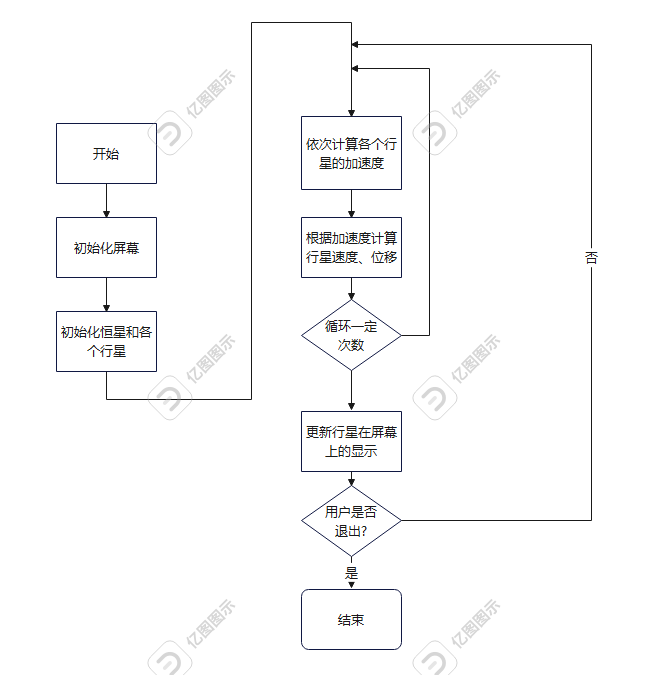
代码中Star类的gravity方法计算行星自身受到的加速度，以及列表中位于自己之后的行星受到自身引力的加速度。

通过修改程序中的G、d\_t、speed等常数，可以自定义引力模拟的效果。

**三、编程技巧说明:**

程序使用Python内置turtle模块绘制图形；使用了面向对象编程的方法，类Star对应一个天体，并继承自内置turtle模块中的turtle类。

程序流程图如下：



程序首先初始化屏幕、恒星和各个行星，然后先计算行星受到重力的加速度，再计算速度和位移。代码如下：

def gravity(self):

# 计算行星自身受到的加速度，以及列表中位于自己之后的行星受到自己引力的加速度

for i in range(lst.index(self)+1, len(lst)):

p=lst[i] # 另一个行星

dx=p.x-self.x

dy=p.y-self.y

d = math.hypot(dx,dy) # 相当于 (dx\*\*2 + dy\*\*2)再开根号

f = G \* self.m \* p.m / d\*\*2

# 将力正交分解为水平、竖直方向并计算加速度

self.ax+=f / self.m \* dx / d

self.ay+=f / self.m \* dy / d

p.ax-=f / p.m \* dx / d

p.ay-=f / p.m \* dy / d

def step(self):

# 计算行星速度、位移

self.dx += d\_t\*self.ax

self.dy += d\_t\*self.ay

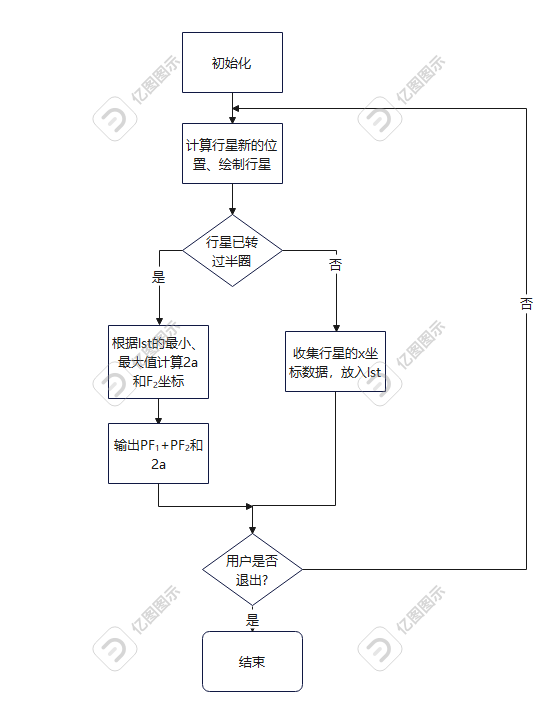
self.x+= d\_t\*self.dx

self.y+= d\_t\*self.dy

**验证开普勒第一定律部分**

开普勒行星运动第一定律，指的是每一行星沿各自的椭圆轨道环绕太阳，而太阳则处在椭圆的一个焦点上。

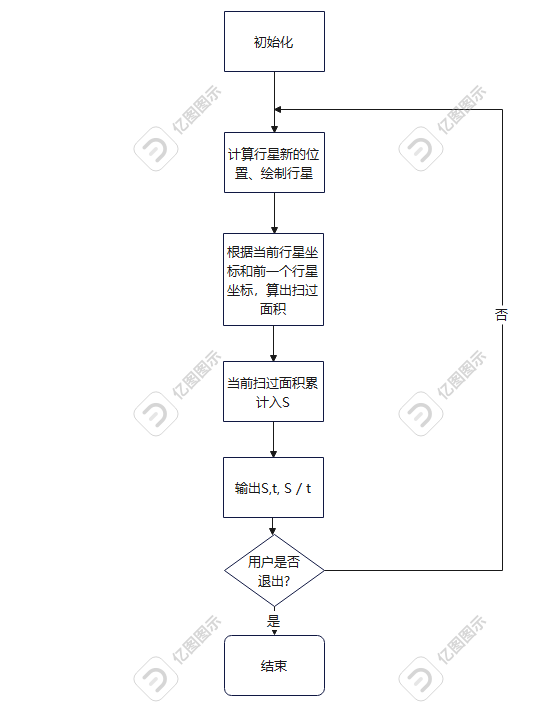
流程图如下：



椭圆上的任意一点到两个焦点的距离之和为一个定值，等于它的长轴。根据这个特性，设P为行星，F1、F2为焦点，太阳位于F1上，如果PF1+PF2近似等于长轴2a，则验证通过。

**验证开普勒第二定律部分**

流程图如下：

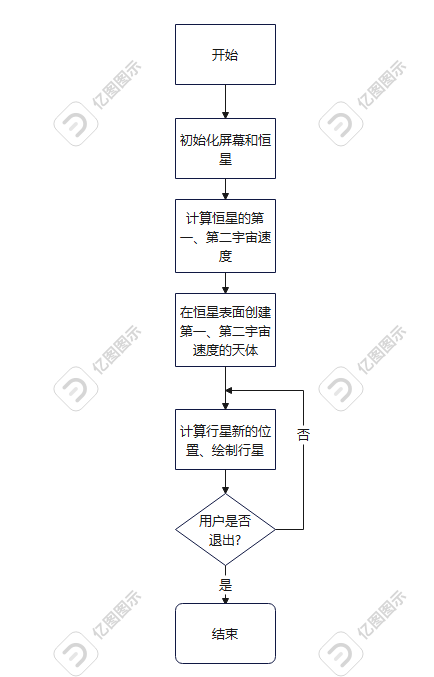


开普勒行星运动第二定律，指的是太阳系中太阳和运动中的行星的连线（矢径）在相等的时间内扫过相等的面积。

程序将行星轨道扫过的部分分割成一个个三角形, 分别计算每个三角形面积，再累加得到行星轨道扫过的面积S, 并除以经过的时间t。如果S/t是一个定值, 则验证通过。

**验证第一、第二宇宙速度部分**

流程图如下：



程序首先计算恒星的第一、第二宇宙速度，然后在恒星表面创建第一、第二宇宙速度的天体。观察现象。可以看到，第一宇宙速度的天体沿着圆形轨道环绕恒星运行，而第二宇宙速度的则沿着抛物线飞出。

**四、参考与引用说明：**

程序中先计算再绘制的循环，以及G、d\_t常量的设计等，有一部分借鉴了标准版Python 3.7中的turtledemo.planet\_and\_moon模块的源码。

另外, 程序Star类的设计，部分借鉴了其中的面向对象编程思想。

**自己创新点：**

1. 计算引力、加速度、速度和位移的部分由本人原创，作者用自己的知识重写了其中的代码。
2. turtledemo.planet\_and\_moon模块中计算一次位移，就重新绘制一次行星，降低了性能。程序对这一缺陷加以改进，计算多次 (次数是speed) 位移，再重新绘制一次行星，提高性能。
3. 加入了原创的验证开普勒第一、第二定律，验证第一和第二宇宙速度部分。

如何看turtledemo.planet\_and\_moon模块的代码？

方法1.打开标准版Python 3.7的IDLE，“Help”菜单 –》 “turtle demo”，“Examples” –》“planet\_and\_moon”，即可。

方法2.打开标准Python 3.7的安装目录，找到文件“Lib\turtledemo\planet\_and\_moon.py”。