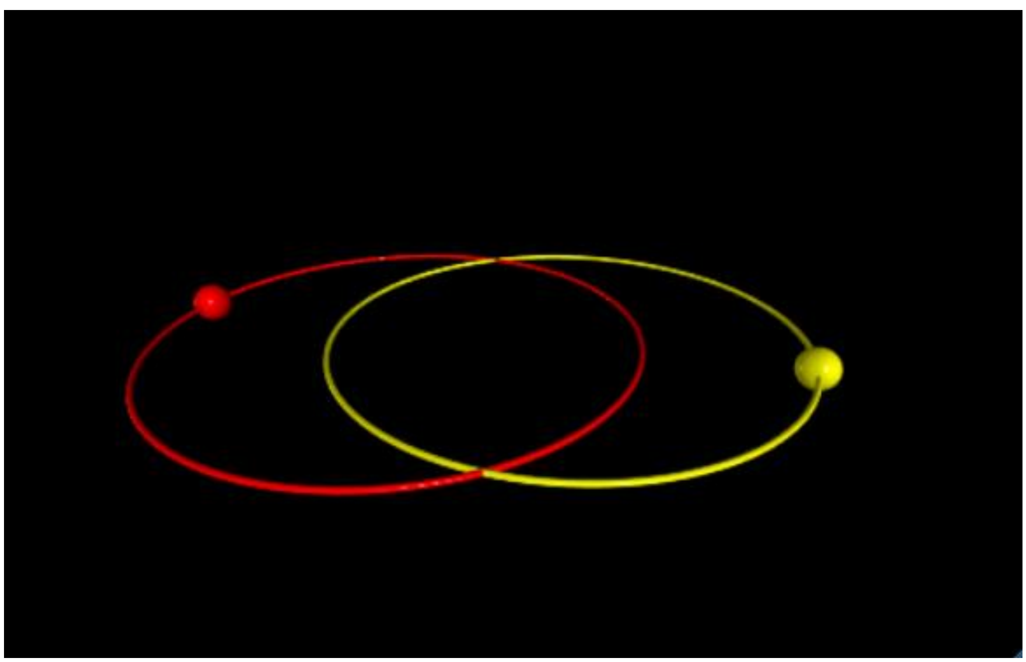
Project: Binary Star System on V Python

Problem statement:

Simulation of a binary star system, a system of two stars rotating around one another, using Newton's law of universal gravitation F=G*m*m/r^2. In binary system both stars move in elliptical orbits about the centre-of-mass (COM).

Solution:

In a simulation of the binary stars system, two stars with some initial velocity move around each other. In the simulation, the viewer will be able to observe two stars that are moving around and acting forces on each other.



Procedure:

We will use an online platform of Glow Script 3.0 V Python on trinket.io for simulation. Further elements and properties for this simulation are described below.

Animation:

For animation, we will employ a **while** loop till the collision between an object and the horizontal level, represented here by ground.

Objects in simulation:

Spheres of suitable radius, named star1 and star2, as an object of motion.

Variables:

Position of the ball, velocity (vector and magnitude) of the ball, acceleration on the ball, angle of projectile. Timestep **dt**, is set to **1e5**.

Values:

We will have found and set the following values as suitable initial values.

star1.mass = 2e30

star2.mass = 1e30
star1.momentum = vector(0,0,1e4) * star1.mass
star2.momentum = -star1.momentum
momentum of star2 is set opposite to star1 so that it also moves around a fix center of mass.
Sphere.pos of star1 = vector (-1.9e11, 9, 0)
Sphere.pos of star2 = vector (1.3e11,9,0)
Position and momentum are the only two values that will be updated inside while loop.

<u>EQUATIONS:</u>

\# Calculate distance vector between star1 and star2 by subtracting their positions.

r_vec = star1.pos-star2.pos

\# Calculate magnitude of distance vector.

r_mag = mag(r_vec)

\# Calcualte unit vector of distance vector.

r_hat = r_vec/r_mag

\# Calculate force magnitude which is equation of gravitational force.

force_mag = G*star1.mass*star2.mass/r_mag2**

\# Calculate force vector by combining magnitude and direction into vector.

force_vec = -force_mag*r_hat

\# Update momenta.

star1.momentum = star1.momentum + force_vec*dt
star2.momentum = star2.momentum - force_vec*dt

Update positions.

star1.pos = star1.pos + (star1.momentum/star1.mass) * dt*1.4
star2.pos = star2.pos + (star2.momentum/star2.mass) * dt/1.4

Code block

The following code was used to carry out the simulation results shown in the results below.

```
GlowScript 3.0 VPython
scene.forward = vector(0,-.5,-1)

G = 6.7e-11 # Newton gravitational constant

# giving position, mass and momentum to both stars.

star1 = sphere(pos=vector(-1.9e11,9,0), radius=2e10, color=color.yellow,
              make_trail=True, trail_type='curve', interval=10)  # positon of star1
star1.mass = 2e30  # mass of star1
star1.momentum = vector(0,0,1e4) * star1.mass  # momentum of star1

star2 = sphere(pos=vector(1.3e11,9,0), radius=2e10, color=color.red,
                make_trail=True, interval=10)  # position of star2
star2.mass = 1e30  # mass of star2
star2.momentum = -star1.momentum  # momentum of star2 is set opposite to star1 so that
                                  # star2 also move around a fix center of mass.

dt = 1e5 # 1e5 for good results
while True:
    rate(500) # speed of both stars changes by changing the rate
    # Calculate distance vector between star1 and star2 by subtracting their positions.
    r_vec = star1.pos-star2.pos
    # Calculate magnitude of distance vector.
    r_mag = mag(r_vec)
    # Calcualte unit vector of distance vector.
    r_hat = r_vec/r_mag
    # Calculate force magnitude or equation of gravitational force.
    force_mag = G*star1.mass*star2.mass/r_mag**2
    # Calculate force vector by combining magnitude and direction into vector.
    force_vec = -force_mag*r_hat
    # Update momenta.
    star1.momentum = star1.momentum + force_vec*dt
    star2.momentum = star2.momentum - force_vec*dt
    # Update positions.
    star1.pos = star1.pos + (star1.momentum/star1.mass) * dt*1.4
    star2.pos = star2.pos + (star2.momentum/star2.mass) * dt/1.4
```

RESULTS

The distances between each star and the centre-of-mass (COM) change with time. When both stars begin to move they start to moving around each other. At all times, the stars are on opposite sides of centre-of-mass (COM) so they never collide.

The simulation of the code is given below.

https://trinket.io/glowscript/168ceeb7cb?outputOnly=true