Double star system

This is a basic exercise with vpython before you can dive into your own quantum or whatever physics experiment simulation. The idea is to simulate a 3 body system consisting of 2 suns with a planet. We use a very simple model: sun1 has a mass 1.0, sun2 has a mass of 0.8. The planet has a mass 0.001. For convenience take the following constants:

R=1, M=1, G=100. Take different colors for the suns and draw spheres for them with radius R. S1.m = M and S2.m = 0.8M and P=0.001 Take a certain momentum: S1.p = vector(0,2,0)*S1.m and P=vector(0,3,0)*P.m S2.p = -(P.p+S1.p) Initialize time to zero with some delta t=0.001 Now in a while loop calculate the influence on eachother:

r12 = S1.p - S2.p
r1p = P.p - S1.p
r2p = P.p - S2.p
F12 = -G*S1.m*S2.m*norm(r12)/mag(r12)**2 etc. t= t+dt Don't forget to put the rate in the loop. It should look more or less as two suns circling around eachother and the planet at a certain distance which is influenced by the sun masses.

Feel free to experiment with something where you are more familiar. First some of the equations:

$\vec{F}_N(\vec{r}) = -\frac{G_N m M}{r^2}\frac{\vec{r}}{r}$

with $G = 6.67\ 10^{-11} m^2 kg^{-1}$ m the mass of the planet, M the mass of the sun. $M = 1.99\ 10^{30} kg$. r = is the distance between the planet and sun

---

```
# planets.ipynb
from vpython import *
import numpy as np

R = 1
M = 1
G = 100
....
S1 = sphere(pos=vector(4.5*R,0,0), radius=R, color.....
S2 =...
P =
S1.m = M
```

```
S2.m = 0.8M
..
t = 0
while t < 500:
   t = t + dt
   F12 = ...
   F1P =
   F2P =
 ...update the momentum and positions of all systems
```