Code for finding the next position, velocity and acceleration
of each body.

```
dT = 5 seconds


//find new positions and velocity of sun and planets
//----------------------------------------------
for (int i = 0; i < nrPlanets; i++)
{
        V[i] = V[i].add(A[i].mult(dT));
        P[i] = P[i].add(V[i].mult(dT));
}

//find new accelerations of sun and planets
//----------------------------------------
A[0] = accSun();
for (int i = 1; i < nrPlanets; i++)
{
   A[i] = accPlanetSun(i).add(accPlanetPlanets(i));
}

--------------------
```

```
//acceleration of planet i by Sun
private Vector3 accPlanetSun(int i)
{
    Vector3 v = V[i].add(V[0]);
    Vector3 rr = P[i].sub(P[0]);
    double rm = rr.mag();
    Vector3 ru = rr.unit();
    double vm = v.mag();
    Vector3 vu = v.unit();
    double m = planet[0].M;

    return ru.mult(-(G*m/(rm*rm))*(1-(4*G*m/(rm*c*c))+vm*vm/(c*c)))
                .add(vu.mult(ru.dot(vu)).mult(4*G*m*vm*vm/(rm*rm*c*c)));
}

//acceleration of Sun by all the planets
private Vector3 accSun()
{
    Vector3 a = new Vector3();
    for (int i = 1; i < nrPlanets; i++)
    {
        a = a.add(accPlanetSun(i).mult(-planet[i].M/planet[0].M));
    }
    return a;
}

//acceleration of planet i by all the other planets
private Vector3 accPlanetPlanets(int i)
{
```

```
        Vector3 v;
        Vector3 u;
        Vector3 a = new Vector3();
        for (int j = 1; j < nrPlanets; j++)
        {
            v = P[j].sub(P[i]);
            u = v.unit();
            if (i != j) a = a.add(u.mult(G*planet[j].M/(v.mag()*v.mag())));
        }
        return a;
    }
```