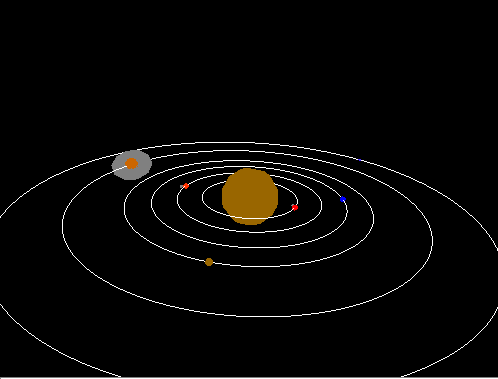
Solar System 

**Description:**

You now are familiar with transformations and the order of operations that must take place. You now must understand how to build a hierarchy of objects that can inherit properties for your scenegraph. In this lab, we will be building a solar system and understand how transformations are carried from one object to another.

**Your Task:**

* You will build a sun, a few planets, and a few moons(or rings) that orbit the planets.
* Understand how to push and pop onto the matrix.

**Files Given:**

main.cpp – You do not need to modify this

solarSystem.cpp – You will write the render function for the solar system.

**Compiling: (On the Mac)**

g++ -Wall -Wextra main.cpp solarSystem.cpp -I/Library/Frameworks/GLUI.framework/Headers/ -framework OpenGL -framework GLUT -framework GLUI -o solar

**Running:**

./solar

**C++ Refresh -- Helper functions:**

glPushMatrix/glPopMatrix

1. #include <fstream>

Local verus Global Operations

1. #include <stdio.h>

**Going Further:**

Did you enjoy this in class assignment?

* Try adding alpha blending to the planets rings. Start looking into textures and other materials that can make the planets appear more interesting.
* In the future when you learn about shaders, you can make the planets look even cooler—feel free to browse the web for more.
* Add satelites that can orbit the planets
* Add asteroids that orbit the solar system
* Create multiple solar systems that all rotate around a galaxy
* Add some interesting simulation
  + If a moon gets too close to a planet, will it get sucked into another planets gravitational pull and rotate about it?
* Add more planets with irregular orbits
  + Pluto for example has a much more egg shaped orbit