COMPUTER SYSTEMS RESEARCH
Running Version of Your Program

1. Your name: Brian Tubergen Period: 2
2. Date of this version of your program: 10/27/08
3. Project title: Creating 2-D and 3-D models of the Solar System using physics-based geometries in Java
4. Describe specifically what files are needed and the command(s) necessary to run your program

```
How to run your demo:

  Need the Sprite class (Sprite.java), Animate01_modified class
  (Animate01_modified.java), and the Point3D class (Point3D.java).

  After that, it's just a matter of compiling and running.
```

5. Your program is running, now what? List test input(s) for the user to interact with your program. Specifically what should the user expect to happen?

```
  Input(s):

  None yet. Hopefully in the future will allow users to click somewhere
  on the screen and create a solar body at the location (and see what the
  system's reaction is to it).


  Program's expected response(s):

  Right now the program just creates solar bodies at locations
  predetermined in the code (the locations have no significance, at the
  moment). For now, 10 pixels corresponds to 1 AU (= 149.6*10^9 m), so
  the planets are a long way from the sun and moving very fast. The
  planet's masses are practically negligible compared to the mass of the
  sun and thus for now I've made the masses so small they essentially
  don't interact with each other (to avoid strange bugs that I won't have
  time to fix before I submit this). Orbits can be elliptical,
  hyperbolic, or parabolic depending on a planet's initial position and
  velocity.
```

6. What about user input errors? Are there incorrect user input(s) that your program handles?

```
  Not yet.
```

7. What is the programming doing, demonstrating, or analyzing? What is the user looking for in order to understand what you've been studying and developing with this project?

   The program is demonstrating basic Keplerian planetary motion; the user should (for the moment) recognize elliptical, hyperbolic, and parabolic orbits of the planets, although in a final version the orbits will appear more circular (after all, planetary orbits *do* appear fairly circular).

   Additionally, I could do real time comparisons with data gathered from NASA or equations that generally predict Keplerian motion.

   Finally, in later stages the user will have the opportunity to put a solar body at a location, and the user will then be able to see what the reaction of the solar system is to that body's presence/motion.

8. How has your program evolved during third quarter to now, the beginning of fourth quarter?

   I'll assume that the question ought to ask how my code's changed from the beginning of the year until now. I had nothing at the beginning of the year; now I've implemented what appears to be a generally working simulation for gravitation, and with correct initial data for position, mass, and velocity, should be able to make it represent the actual Solar System without a ton of difficulty.

9. By the end of this school year, what do you hope to have as a final version of your program in relation to this current version? What will you demonstrate during your final presentation?

   Hopefully I'll have a working model of the solar system that users can interact with in the previously mentioned way. Hopefully the program will also be in 3-D, although because I realize this will be difficult, I'm striving to create a good 2-D simulation first.