**CSCE 155N Matlab Programming Project 1 – Summer 2014**

**Assigned: Tuesday 7/21/2014**

**Due: Friday 8/1/2014 at 11:59 PM**

**(hardcopy in class on Monday)**

**Into Lunar Orbit!**

**Problem Statement:**

This is a simulation project. Your program should allow a user to visualize and modify the orbit of a satellite around the moon. There are minimum expectations, plus many enhancements that you are very welcome to implement for bonus credit. (Bonus points or “stars” accumulate through the semester separate from regular credit points. After preliminary final grades are calculated, these bonus stars are considered for adjustments of generally half a grade and sometimes more.) See the project rubrics for suggested enhancements.

Features for a minimal implementation include the following:

* Visualize a moon (simple sphere) on a plot.
* Visualize an orbit and location of the orbiting object in an animation.
* Provide for pauses so that a user can set a thrust vector that can modify the orbit.
* Use realistic values for all parameters, and show scaling factors for speedup etc.
* Use standard formulas for orbital mechanics, including a G which varies by distance from the moon.
* Include complete documentation as detailed below and in the rubric.
* Include evidence of testing in an incremental, modular fashion.

You are to work in teams of two or three students to design prototypes in Matlab. Larger teams may be allowed by permission, but there would be higher expectations. Teams should include at least one more “advanced” student and one “beginner” student.

**Resources:**

There are many online resources. For Matlab, certainly check out <http://www.mathworks.com/help/matlab/index.html#graphics> and expand out the Contents to find extensive help with 2-D and 3-D plots. Much of the mathematics and terminology for orbital mechanics can be found at <http://www.braeunig.us/space/basics.htm>. Simulating orbits using Matlab is the topic of a forum at <http://forum.nasaspaceflight.com/index.php?topic=15063.30> . Make use of your Piazza account to ask for and share research sources. Do remember that it is very important to document exactly where you found materials and code samples. It is expected that this project is ultimately yours.

**Collaboration:**

Work together as a class on any or all aspects of the research and design. Ideally take advantage of the talents of each member of the team, but recognize that each is responsible for the entire project! This means being prepared to answer questions on the code even on parts you did not program. It is essential to keep track of who did what and where any useful information was found. Record each time you helped someone else and each time someone helped you. Keeping a log is highly recommended. Note that Piazza keeps a record automatically!

**What and How to Submit:**

Read and have your program conform to the “Program Documentation Guidelines” which are online.

By the deadline hand in electronically the two files, lunar.m (the Matlab script file for the simulation), and lunar.docx (which contains summaries, documentation, and sample runs). Also submit a hardcopy of a grading rubric form. Only one copy of everything per group needs to be submitted – it does not matter who does the handing it. In class the day after the deadline, hand in the hardcopy version, stapled together with the cover page in front.

Each team member should electronically on his/her own account submit his/her own analysis of the relative contributions of all the members toward the project. This is in addition to the acknowledgement section of the main report. Assuming allocation is fairly even, all will receive the same grade.

The Word document should contain the following, all carefully labeled:

* Cover page with name(s) and the account under which it is submitted, title, date submitted, etc.
* A discussion of the features you implemented in the project. Describe how they work and what Matlab options were used to program them. This should be at a fairly high level, not a line-by-line analysis of the code.
* Acknowledge all collaborations (both internal to the team and external), detailing what each person contributed individually, and what was done jointly. Indicate approximate percentages of the work contributed by each person in design, coding, testing, documentation, and report preparation.
* A discussion of the testing that was performed. This should include testing of each component as it was being built, and testing of the final program ensuring that it works properly under a comprehensive range of conditions. Printouts may be useful
* An “instruction manual” that a non-programmer can use to run the duel game.
* A copy of the code