**CSCE 155N Matlab Programming Project 1 – Summer 2012**

**Assigned: Tuesday 7/17/2012**

**DUE DATE:**

**Wednesday 7/25/2012 at 11:59 PM**

**(hardcopy in class on Thursday)**

**Get It Before It Gets You!**

**Problem Statement:**

The class Web site has a short script *cannonGame.m* that has a single player fire cannon shots at some stationary object with the goal of hitting it in a minimum number of shots. A plot is displayed showing the arc of the cannonball.

Your task is to expand on this simple game. At a minimum let’s have the target move toward your player at a constant velocity so that there is only a limited time to shoot it dead. This means that your program will need to keep track of the time. Let the time begin when you fire your first shot. This could be interpreted as the attacker figures out where you are when seeing the smoke rise from your cannon.

Have the user enter his/her name so that the computer uses it in the updates. Then rather than settle for a cannon, allow the user an option of several weapons. Research and use realistic data for the weapons you permit, such as speed, weight (if relevant), maximum kill distance, maximum injure distance. Similarly, let the user select from several possible targets. If the target becomes injured, have it continue moving at a slower speed.

Additional features (that garner more points) include 3-D plotting, user-selected charge percentages, multiple targets, more random options, etc.

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.

**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 if your allocated task was to write the report. 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.

**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, cannon.m (the Matlab script file for the game), and cannon.doc (which contains summaries, documentation, and sample runs). Only one copy 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.
* An “instruction manual” that a non-programmer can use to run the cannon game.
* An annotated cut and paste sample dialog of the running of the program. (Hint: Use the ‘diary’ command and/or capture screen shots.)
* 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.
* An annotated cut and paste of a sample dialog, demonstrating how your program responds to extreme and faulty input. (This could be combined with the previous section.)
* 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.

**Grading Criteria:**

* Program functions as intended – 30%
* Program logic is well designed – 20%
* Documentation guidelines are followed – 20%
* Handin Documents formatted and arranged as specified – 10%
* Testing is comprehensive – 10%
* Quality of the user’s manual – 10%