EECS 338 – Final Project

General instructions: The final project is worth 10% of your grade for the course. You will write an application of your own choosing. You can work by yourself or in a group of 2 people. The project must agree with the accepted proposal, and major design changes after the proposal must be approved. The workload is expected to be approximately 20 hours per person. For a group project, both students will generally receive the same grade, except in cases of unequal effort. The project must focus on one or more of the following topics regarding operating systems from each major section of the textbook:

1. Process management
2. Memory management
3. Storage management
4. Protection and security

The exercises and programming projects in the textbook may provide ideas. Other relevant topics may be permitted if approved *in advance*. Projects can be developed on personal computers, but they must also be compiled and demonstrated on either an EECS server or the Case HPCC. You may use any programming languages you wish.

Development Phases (see due dates on websited):

1. *Proposal (10 pts)*: Complete the online form (see class website for link). You will be graded based on clarity and completeness. The design of your application can be changed later, but any significant changes should be approved by the instructor and accompanied by a revised proposal. If you are in a group, only one submission is required.

2. *Beta Version (30 pts)*: In the software industry, "beta" indicates a version that is ready to be tested outside of the company (http://en.wikipedia.org/wiki/Software_release_life_cycle). Yours is allowed to be quite worse than a typical beta. Your submission should include the following:
   a. (9 points) A typed design document that provides a description of all files, major data structures, and sample output (console or data file). Grade will be based on clarity and completeness.
   b. (7 points) All necessary files, including program and data files. These are not required to be functional, but comments should be inserted where necessary to explain significant parts of the program. Grade will be based on completeness and agreement with the design document.
   c. (7 points) All major data structures such as arrays, synchronization variables, etc. These are not required to be used, but they should be declared in the appropriate places. Grade will be based on completeness and agreement with the design document.
   d. (7 points) Demonstration of any output (console or data file). This can be "mock" output that is generated artificially. Grade will be based on completeness and agreement with the design document.
   e. For a 2-person group, include a typed explanation of what each group member worked on. If missing, 5 points will be deducted.

3. *Final version (60 pts)*: This submission should be complete. Be sure to simplify the application if your original ideas are not working. Include a typed design document that provides a description of all files, major data structures, and output (console or data file). *Note: do not use more than 10 MB of support files because these may be difficult for teaching staff to download.* Grade will be based on the following criteria:
   a. (20 points) Design document is clear and complete (see requirements above).
   b. (20 points) Meets primary, approved objectives from proposal.
   c. (20 points) Basic operations can be replicated by grader.
   d. For a 2-person group, include a typed explanation of what each group member worked on. If missing, 10 points will be deducted.

4. *OPTIONAL: Presentation (+10 pts)*: This part is optional and is equivalent to 1% of your grade for the course. Either demonstrate your program or present a design plan describing its features and showing at

least one screen shot (actual or hypothetical). Presentations should be approximately 5 – 10 minutes in duration. A signup form will be provided, and advanced registration will be required.