ECE485/585 Homework/Project Grading Rubric

The following rubric will be used in grading all homework and project assignments for this course. There are five categories under which your programs will be graded including functionality/specifications, readability, documentation, efficiency, and testing/demonstration. Each category will be assigned a specific rubric score (0-4). Rubric scores will be multiplied by a weighting factor for each category to determine the final score for the assignment.

Functionality/Specifications (35% of total score)	Rubric Score
The program is completely functional and responds correctly producing the correct outputs and or responses under all test cases.	4
The program is mostly functional and responds correctly producing the correct outputs and or responses under most test cases. There are minor problems with the program implementation.	3
The program is marginally functional with numerous errors. The program may respond correctly under certain circumstances, but there are significant errors and/or incomplete program sections.	2
The program is minimally functional with significant portions of the program missing or incomplete. The program is largely nonresponsive to most test cases and/or inputs.	1
The program is not functional, meeting no significant design specifications, or was not attempted.	0

Readability (25% of total score)	Rubric Score
The program is extremely well organized, properly formatted, and easy to follow. Related code sections are logically grouped.	4
The program is reasonably easy to read with logical groupings of instructions. There are minor formatting problems.	3
The program is readable only with significant effort. There is little to no formatting and/or significant problems with code organization.	2
The program is poorly organized and difficult to read. There is little to no consistency in formatting and logical code grouping.	1
The program is readable only by the author or someone extremely knowledgeable with its layout and purpose.	0

Documentation (20% of total score)	Rubric Score
The program is extremely well documented. All rungs and instructions are appropriately labeled with meaningful names and descriptions. Comments are completely consistent with the associated rungs and/or instructions. There are no grammar or spelling errors.	4
The program is reasonably well documented. There are minor formatting omissions that would have improved user understanding of code purpose. There may be limited grammar or spelling errors.	3
The program is marginally documented. There are significant portions of the code that are not documented or documented incorrectly. There are a significant number of spelling and/or grammar errors that detract from the documentation.	2
The program is poorly documented. There are minimal comments and/or the comments are incorrect for the associated rungs and/or instructions.	1
The program is not documented.	0

Efficiency (10% of total score)	Rubric Score
The code is extremely efficient without sacrificing readability and understanding. Instruction addressing is logical and consistent throughout the program. When appropriate, the code makes proper use of subroutines.	4
The code is mostly efficient without sacrificing readability and understanding. Instruction addressing is logical and consistent throughout the program. Some improvements could be made through a better choice of instructions and/or subroutines where appropriate.	3
The code is marginally efficient. There are a significant number of cases where use of different instructions should have been considered. The approach used in implementing the program leads to inefficiencies.	2
The code is a brute force implementation and unnecessarily long.	1
The code is inappropriately long and appears to be patched together.	0

Testing/Demonstration (10% of total score)	Rubric Score
The code has been completely and thoroughly tested prior to submission to the instructor. Documentation is provided concerning all tests performed and the results of the program execution in those test cases.	4
The code has been mostly tested prior to submission to the instructor. A few test cases have not been implemented or considered and/or the results of the tests are not documented. Additional testing by the instructor reveals limited incorrect operation not previously discovered.	3
The code has been marginally tested prior to submission to the instructor. Many test cases have not been implemented and/or the results of the tests are not documented. Additional testing by the instructor reveals significant incorrect operation not previously discovered.	2
The code has been tested only minimally prior to submission to the instructor. Most test cases have not been implemented. The results of the tests are not documented or are misleading with respect to program operation. Additional testing by the instructor reveals major incorrect operation not previously discovered.	1
The code has not been tested prior to submission.	0