Guide to Program Requirements and Grading

Programs are graded based on the following criteria.

**Program Documentation (10%)**
Comments explaining the various logical sections of your program are required. A "logical section" may be an `if/else` statement that is making an important decision, a `while` loop that is a component of the program, or any place code whose function is not obvious.
The reason for these comments is to explain to other readers of the program why the statements that follow are necessary. Therefore, it is important to explain **why** they are there and not **what** they do (since a Java programmer is reading the program, there is no need to explain what Java code does.)

**Javadoc Documentation (10%)**
Javadoc provides a means of creating documentation in HTML format, and should be included at the beginning of a program or class to explain its basic function, and to document all the public constants and methods so other users of the class have a clear understanding of what the methods do.

**WhiteSpace and Indentation (10%)**
Again, the ability to read a program is directly related to the time it will take another user understand the code. Consistent indentation is expected using space characters and **never** tab characters. Most editors, including Eclipse, can be configured to insert an appropriate number of space characters when the TAB key is pressed.

**Program Logic (30%)**
Your program's code should make sense. Not that we are expecting the most efficient or elegant code, but it should not be clumsy and it should conform to the techniques show in lecture that were taught to you to develop the program. This is not to discourage experimentation with alternative code choices, but you must be ready to defend and explain why they were chosen.

**Program Execution (40%)**
Clearly, your program should run to a normal completion and behave as the requirements given specify. Partial credit is given for programs that do not fulfill all the requirements. Credit is **never** given for a program that will not compile.