Programming Rules/CSCI 235

Read this so that you will not be surprised later. The programming component of this course is worth 40% of your grade. I am willing to point you in the right direction by answering any questions regarding your assignments, do not hesitate to send me an email. It is extremely important to follow the following rules. Failure to follow them will result in a lower grade in your assignment, and in many cases could result in zero credit.

Read the following carefully. Please follow the following rules (no exceptions!):

1. All assignment submissions will happen through email (sb1037@hunter.cuny.edu). You will be submitting the source code and other documents as a single archive (.zip or .tar.gz) as described below. To submit an assignment you must submit electronically via email a zipped (.zip or .tar.gz) directory that contains the following (do not submit .rar archives):
    a) the program source code (all source and header files). It is important that each source file starts with a commented header that includes your name.
    b) your Makefile. The use of a Makefile is mandatory. All programs should compile by typing "make all". Failure to provide a Makefile will result in a 20% penalty in your assignment grade.
    c) a README.txt file in which the following are explained:
        i. Which parts of your assignment were completed.
        ii. Any bugs that you have encountered.
        iii. Complete instructions of how to run your program(s).
        iv. The input file (if any) and the output files (if any).

2. IMPORTANT: Please make sure that the name of the submitted directory is your full name. For instance name the directory as John_Adams_AssignmentX.
3. Your program should compile and run correctly in one of the Linux machines of 1000G Lab at Hunter North. All lab machines have the same configuration with the g++ compiler. You should have an account in the 1000G Lab. Note, that you can access the lab machines remotely.
4. Also note that we will not debug your program. We are not going to alter your code in any way. Your code should compile and run as submitted. For full credit, an assignment must be submitted no later than midnight on the day it is due.
5. For each late day it loses 15% of its value. After three late days the assignment will not be accepted. The program must be your work and your work alone. Attributing someone else's work as your own is plagiarism, which is not only a violation of College policy, and is not only an unethical activity, but is also a crime. Your programs will be checked with turnitin.com and with other screening software in order to prevent contract cheating. Contract cheating is a form of academic dishonesty in which students get others to complete their coursework for them. Please read more information on Contract cheating from http://en.wikipedia.org/wiki/Contract_cheating. Please note that you can use the source code associated with the text-book unless otherwise mentioned in the assignment. You can exchange ideas and suggestions with your fellow students, but you are not allowed to copy code.
6. The grade of your assignment will be based on the following:
    a) Every program must be correct to receive full credit. "Correct" means that for every possible input, it produces output that is consistent with the specification. If the program produces correct results for some, but not all, inputs, it is not correct. Since there may be infinitely many possible inputs, you cannot possibly establish your program's correctness by running it on all inputs. You must use a combination of sampling (i.e., testing) and logical analysis to convince yourself of its correctness. Correctness is usually 50% to 60% of the grade. Every program must satisfy specified performance requirements. This means that it uses an amount of storage and running time within specified or reasonable limits.

b) Every program must be well-designed. This is a subjective criterion. There is no one way to design a program well. There are, however, commonly accepted standards of what "well designed" means. These include :

   i. using appropriate ADTs and algorithms,

   ii. decomposing the program into appropriately-sized modules that are cohesive,

   iii. that decisions about which objects are exposed and which are hidden are justified by sound arguments,

   iv. that classes are trim but adequate in terms of the methods they provide,

   v. that there are no global or static variables unless the problem cannot be solved without them, and

   vi. that the design corresponds to the problem statement in a way that can be explained in the documentation.

   There are no rigid rules. All rules can be broken, but it is best to check with me before taking a big chance. A program's design is worth from 10% to 25% of the grade.

c) Every program must be professionally documented:

   i. Every distinct source code file must contain a preamble with the file's title, author, brief purpose, date of creation.

   ii. All functions must have a prologue containing comments for each parameter and appropriate pre- and post- conditions.

   iii. All non-trivial algorithms must be documented in plain English in a multi-line comment block.

   iv. All non-trivial declarations must have adjoining, brief comments.

d) Documentation is worth 10% of your grade.