Project 3 Generic Grading Guidelines
COP 3014
Total possible points: 100
Due Time: As Announced in Project Write-Up
Turn in "ON TIME": by 11:59 pm on due date

Note: these are generic grading guidelines, intended to give you an idea of how the project will be graded and what kinds of things the grader will be looking at. These guidelines are tentative and are subject to change. You will be provided with specific guidelines when you get your graded project back. Before you turn in your program, be sure to read the style guidelines handout on the course web site and utilize all information you have been given in lecture, recitation, the textbook, via email, and on the web site.

Working program/Correct Results: worth 50% of total points

- syntax errors - does not compile (up to -50% penalty)
- syntax warnings, as relevant
- compiles but will not run (link errors etc.)
- unreachable code (e.g. if/then always false)
- errors in expressions, calculations and equations
- missing evaluations for special cases
- not reading input data correctly
- did not echoprint input, if required
- missing output which was required
- runtime error (program crash)

Documentation/Style : worth 35% of total points

Use the course Programming Style Guidelines for C++ handout as a guide for requirements of style, formatting, documentation, etc.

- shortage of documentation / header comments
- bad output format esthetics (e.g. no white space, meaningless output messages, unlabeled output)
- bad output variable formats (e.g. unformatted numbers)
- lack of comments explaining code
- lack of comments describing variables, constants, etc.
- lack of comments on { } pairs (where needed)
- inaccurate or meaningless comments
- insufficient white space in code (around lines, sections, operators etc.)
- meaningless or overly-abbreviated identifiers
- did not declare appropriate named (symbolic) constants

Efficiency / program structure: worth 15% of total points

- unnecessary code (each) (e.g. repeating nearly identical code)
- unnecessary conditions on if which cannot be justified
- unnecessary { } s (each)
  - exception: situations where the code structure may be clarified by using an extra { }
- using repeated if/then (instead of if/then/else or switch)
- code inside an if (or switch) that belongs outside
- use of return, break, continue when not justified
- use of goto
- code inside a loop that belongs outside the loop
- poorly organized or unclear program structure

Miscellaneous Notes:

- when output is incorrect, points are taken off for the *cause* of the errors (e.g. errors in expressions, equations, missing equations, etc.)
- don't deduct more points than an area is worth (e.g. don't deduct more than 35% of possible points for documentation etc.)
- points will be taken off for the use of C constructs which are not compatible with the focus of this course, which is for students to learn the conventional use of the standard C++ language (not to learn C). For example, points will be deducted for using scanf or printf instead of cin and cout, for using pointer parameters instead of reference parameters, and for using #define instead of the const qualifier to declare named constants. Students must use only standard ANSI C++ header files which are discussed in the course textbook or in class as being part of ANSI standard C++. Standard C++ header files do include C++ headers derived from C such as cmath.h, but they do not include old C headers such as math.h. Note that files such as stdafx.h and conio.h are NOT standard and cannot be used.

Automatic zeroes:

- not turned in by late deadline
- the following situations, if an attempt at "cheating" appears to have occurred
  - two or more programs which are substantially identical
  - if output is turned in, it could not have come from the program turned in

Late penalties:

Deduct 20 points if turned in by 11:59 pm 48 hours following the original due date

Missing/Extra Item Penalties:

- not submitted electronically and correctly to Blackboard site
- required file missing (this results in a zero score)
- required file not named correctly
- more than one file turned in
- file turned in must compile and run correctly using Microsoft Visual C++ Express 2008 compiler on a Windows machine

CORRECT RESULTS: see sample solution
