

Homework Assignment #5: Programming Project #1

Due Date/Time

2:10PM Monday, November 30, 2015. Late submission will be penalized by 20% for each working day overdue.

Task Description

Develop a C++ application (called “myCalculator” perhaps) that reads a C/C++ arithmetic expression, evaluates it, and prints out the result. The simplest kind of arithmetic expression that you must handle is a C/C++ constant integral expression built up from integers and the five arithmetic operators + (binary), - (binary), *, /, and %, with possible parentheses (and) to group subexpressions. Below are two examples:

- $1 + 2 - 3 * 4 / 5$
- $(1 + 23 - 456) * (78 / 9)$

Be careful with illegal inputs. When the input is illegal, your program should be able to report an error and stop (or ask for another input).

You may go beyond this basic requirement, but please follow the C/C++ syntax and semantics for integral arithmetic expressions.

Submission Guidelines

- Submission should be in printed form.
- Code listing and test cases must NOT be on the same page.
- Please use A4 paper and *double-sided* printing.
- Simply staple on the upper left corner; NO plastic or cardboard covers and NO binders, either.
- Drop your homework by the due time in Yih-Kuen Tsay’s mail box on the first floor of Management College Building II.
- *You may be requested to demonstrate your program.*
- DO NOT plagiarize (i.e., do not use material without crediting the source). You may discuss with others, but copying code is strictly forbidden.

Grading

This assignment constitutes 5% of your grade (of this course). Your work will be graded according to its completeness, correctness, and presentation. You should provide evidences (such as tests) showing that your program is correct. You should also organize and document (by adding comments to) your program in such a way that other programmers, for example your classmates, can understand it. Below is a more specific grading policy:

Criteria	Score
incomplete or doesn't compile	≤ 20
complete, compiles, but with major errors	≤ 40
handles only single-digit integers	≤ 60
complete, but with minor errors	≤ 80
complete and correct	≤ 100
prints out an equivalent postfix expression	+5
can handle the unary -	+5
can detect an overflow/underflow	+5
allows variables and assignments	+10