



North South University
Department of Electrical and Computer Engineering
CSE 225: Data Structures
Fall 2014
Assignment 01

Problem specification:

This assignment is about using the Stack data structure to solve real-life problems; in this case, solving arithmetic expressions written in infix form. You have to write a complete C++ program which takes an infix expression from the user as input, determines the outcome of the expression and gives that back to user as output. You will have to solve this problem in two steps. First, you have to convert the expression from infix notation to postfix notation. You are going to need a stack in order to do so. In the next step, you will have to evaluate the postfix expression and determine the final result. Again, you will need a stack in order to do this. In both steps, you must use the stack class we developed in our class lectures.

Input specification:

Your program must read the input from a file named in.txt. The input begins with an integer N. In the following N lines, N infix expressions are provided, one expression per line. All the operands in the infix expressions are single digit non-negative operands and the operators include addition (+), subtraction (-), multiplication (*) and division (/).

Output specification:

Your program must store the output in a file as well, named out.txt. For each infix expression, your program either generates a single number which is the outcome of the infix expression, or the text “Invalid expression”, if the expression is not a valid one. Do not produce additional outputs like “Please enter expression:” or “The result of the expression is”.

Sample input:

```
4
10 + 3 * 5 / (16 - 4)
(5 + 3) * 12 / 3
3 + 4 / (2 - 3) * / 5
7 / 5 + (4 - (2) * 3
```

Sample output:

```
11.25
32
Invalid expression
Invalid expression
```

Notes:

If you are not sure how to read/write input/output to/from file, you can watch this tutorial.

<https://www.youtube.com/watch?v=gUFJW9Bmu-k>

Submission guidelines:

Let's assume your NSU student ID is 1234567890. After you complete the assignment, zip the entire project folder, rename it with your ID number (in this case, 1234567890.zip) and upload 1234567890.zip on Engrade in the turn-in section.

Any form of cheating will be penalized heavily. Duplicate codes (no matter if full or partial) will not be marked regardless of which one the original is.