CS218 Data Structures

Spring 2020 FAST-NU, Lahore

Instructor: Hafiz Muhammad Hamza

Assignment 3 – Stacks

Develop a C++ program to evaluate arithmetic expressions. The expressions consist of numbers, the four arithmetic operators, one unary operator and parentheses. Implement your own template Stack class for this purpose.

Your program is expected to:

1. Read several arithmetic expressions from a file named *input.txt*
2. Convert an infix expression to postfix expression using Stack
3. Evaluate the postfix expression using Stack

The minimal set of operations that your program should be able to handle is:

- binary: + , – , * , /
- unary: –
- parentheses: ( )

Default associativity rule to be maintained is:

- left to right

Algorithm

Part 1: Infix to Postfix

```
initialize stack
while tokens remain in infix expression
    x = get next token
    if x is operand
        add to postfix expression
    else if x is an operator
        while precedence (top of stock) >= precedence(x)
            pop operator and add to postfix expression
        push x to stack
    else if x is a "("
        push x to stack
    else if x is “)”
        while top of stack != “(“
            pop operator and add to postfix expression
        pop “(“
while operator stack not empty
    pop operator and add to postfix expression
```

Part 2: Evaluate Postfix

```
initialize stack
while tokens remain in postfix expression
    x = get next token
    if x is operand
        push to stack
    else if x is an operator
        if x is binary operator
            pop operand from stack (right operand)
            pop operand from stack (left operand)
            apply operation
            push result to stack
        if x is unary operator
            pop operand from stack
            apply operation
            push result to stack
the last value remaining in the stack is the answer
```