**CS 248 – Object-Oriented Programming and Data Structures**

**HW5: 100 points**

**Objective:**

The objective of this homework is to learn about:

1. Develop knowledge regarding different operations of stack/queue
2. Learn and implement different notations (infix, prefix, and postfix)
3. Learning to use the Stack for different types of objects (Integer and Character)
4. Use of stack in solving real life problems
5. Learning to use Integer and Character objects (boxing and unboxing)

**Problem Statement:** In class, we have learned about the process of converting Infix expression to postfix and how to evaluate such expressions. In this homework, we will write the code that will take an expression (input) from the user in infix notation and convert that to corresponding postfix notation and evaluate its value (bonus part: see below).

**Task 0:** Use the starter code to start. All the needed functions and their functionalities are given in the starter code. Please note, both the tasks (Infix to Postfix and evaluate Postfix should be able to handle the following operations: +, -, \*, /, ^ (power)

**Task 1:** Complete the task for infix to postfix conversion by completing the following functions:

public static int pr(char elem)

public static String InfixToPostfix(String infx)

public static int evaluatepostfix (String pofx)

**Input Format:** You need to read from a file named **in.txt**. First line of the file contains a string in infix notation that needs to be converted in postfix notation.

**Output Format:** Output should be printed in the console.

**Submission Instruction:** Submit only StackArrayInfixPostfix.py through Canvas.

**Special note:**

1. **Consider, there is no space in input/output and all values are single digit (for example: 10 means two single digit values 1 and 0)**
2. **The file names must be exactly the name mentioned. Failure to follow this would result in a 0.**

**Sample Input/Output:**

**File input:**

(4+8)\*(6-5)/((3-2)\*(2+2))

**Output:**

Infix: (4+8)\*(6-5)/((3-2)\*(2+2))

Postfix: 48+65-\*32-22+\*/

Postfix evaluation: 23

**File input:**

3+4\*5/6

**Output:**

Infix: 3+4\*5/6

Postfix: 345\*6/+

Postfix evaluation: 6

**HW Grading:**

1. 5% - Follows style guidelines, including header and correct filename.
2. 45% - Compiles without errors
3. 50% - Correct implementation of all functions

**Bonus (10 points – capped at 100):** Write the Postfix expression evaluation function.