**CSE 3302: Programming Languages LAB**

**RPN (Reverse Polish Notation) Calculator using Python**

**100 points**

**INSTRUCTIONS**

1. **Do NOT plagiarize.**
2. **No group work. All work should be your own.**
3. **Do not discuss your work with other students in the class.**
4. **You CANNOT borrow code from online sources.**
5. **Turn in your program using Canvas. Do not email your program to the TA or the instructor.**
6. **Name your document as [netid]_LAB3.py where netid is your UTA netid. If you do not know your netid, check what it is using NetID Self Service. Your 1000 number is NOT your netid. If your file name is wrong, your assignment may not be graded.**
7. **All code should be your own. You may not copy code from the slides, book, others, or the internet unless specified. You are not allowed to use in-built functions other than the ones taught in class for functional programming.**
8. **There is a 5% bonus available for turning in the assignment 2 days early.**
9. **The code should have you name, 1000 number, the date you turn in your assignment, and OS used as the first 4 lines in order.**
10. **It is suggested to use Python 3.6.2 or later. You can download it for free at python.org. Please let the GTA know BEFOREHAND if you are using a different version.**
11. **You should only need to use the 'os' package, if you want to use any other package you must get permission beforehand.**
12. **Input is RPN**
13. **Output is the calculated result of the expression**

   Use Python to create a simple calculator that accepts Reverse Polish Notation (RPN) and displays the final answer (Intermediate steps or results need not be displayed).

   It only accepts 4 operators "+", "-", "*", "/".
   Input numbers will be single digits.
   The input will be in postfix notation.
   The input will be provided in a text file called **input_RPN.txt**.
   Your program should not ask the user for any input.
   There will be one RPN expression in each line.
   Your code should be able to read the file and print the result for each RPN in a new line.
   Example of RPN: 4 2 + and your output should be 6. This is a simple expression. More complex algebraic notations will be used to test your program like the one below.
   Example algebraic notation: ( 4 + 2 * 5 ) / ( 1 + 3 * 2 )
   Translated into RPN: 4 2 5 * + 1 3 2 * + /

Note: - Your code should be able to read the input file from the same folder (which has your .py file). Do not hard code the path to the file in your laptop/desktop. Use os to get the path and read the input file. Also, please take special care to process the line-ending character correctly; for example, if you write the program on a Mac it should work correctly when graded using Windows and vice versa.

**Extra credit (5 points each)**
A) Write a separate program that can input an algebraic expression and convert it to RPN and then evaluate the RPN. Print the RPN and the result in separate lines. If you are implementing extra credit, your file should be name as **<netid>_EC.py**. The input file name will be **input_RPN_EC.txt** and it will have algebraic expressions.
B) Add at least one more operator (unary subtraction, or modulo division, etc.). You must document what operators you are adding. Add which ones to comments and make sure to include that as well in your submission so the GTA knows to test using the extra scenarios.