Assignment 5
Calculator Compiler

Lex is a lexical analysis tool that can be used to identify specific text strings in a structured way from source text. **Yacc** is a grammar parser; it reads text and can be used to turn a sequence of words into a structured format for processing. In this assignment, you have to develop a desk calculator application using both lex and yacc. The calculator should support BODMAS rules and operators. Standard precedence and associativity rules should be taken care of. Numbers can be integers or floats, positive or negative, so you have to take care of unary operators as well.

Submit both **calc.l** and **calc.y** files. Lex will generate lex.yy.c file which contains yylex() function which is called by yyparse() to get the token. Calc.y will contain grammar rules to parse the input, which you have to formulate with respect to this problem.

Unsuccess:
For any input that does not conform to the grammar rules, the program should print **'syntax error'**.

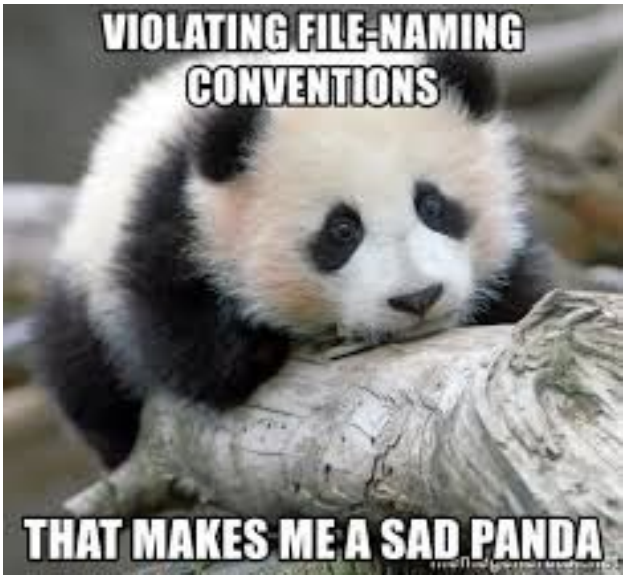
Success:
1) Convert the infix form to postfix.
2) Return the numerical value of the input.

Following is the sample input-output example:

User > (2 - 3) ^ 4 * 5 / 6 + 12 - 9
Calc > Postfix is 2 3 - 4 ^ 5 * 6 / 12 + 9 - **Calc >** Value is 3

Create a zip file named **Roll_NO(All Caps)_a5** containing both the lex and yacc files **(calc.l, calc.y)** and a **shell script** (Compile both the files).

******



!! Happy Coding !!

Assignment 6
A toy C Compiler

Symbol Table is an important data structure created and maintained by the compiler in order to keep track of semantics of variable i.e. it stores information about scope and binding information about names, information about instances of various entities such as variable and function names, classes, objects, etc.

Go through the following link thoroughly.

https://github.com/trilliwon/cminus-compiler

For this assignment you may expect to do some changes in the grammar, add/remove functionalities, change syntax of any statement (for/while/if-else) in the lab itself **Next Thursday**.