Requirements: (DUE 18 December 2017 (Monday), 5 pm)

1. To build an interactive menu driven system with the following functions:
   A. **Convert to infix, prefix or postfix.**
   B. Evaluate any type of expression (infix, postfix, prefix)
   C. Exit

Note: Your program must be able to determine the current expression before converting to the other two types. Example (sample output):

5+3 * (9-2)
This is an infix expression; do you want to convert it to (A) postfix or (B) prefix or (C) for both? Select A, B or C option.

2. User will be given option whether to read from file (user needs to type the location of the file) or from default input (keyboard):
   • If it is read from file, your system should be able to show on screen before any of the functions mentioned is being performed.
   • If it is read from default input, your system should be able to save it in a default file – **input.txt**

3. All outputs are to be shown on screen:
   • If the input is read from file, all outputs (correct conversion and evaluation only) should be appended in the same file.
   • If the input is read from keyboard, all outputs (correct conversion and evaluation only) should be written in a default file – **output.txt**.

4. Your program should only evaluate the following operators:
   • +, -, *, /, ^ (power), (,).

5. Imbalanced parentheses must be handled correctly, example:
   if there is any imbalanced parentheses occurred, your system should have alerted the user – which part with error.

Data sturctures:
6. Only **singly linked list** (pointer) to be used to illustrate stack. Marks will be deducted (-10%) if any other data structure other than singly linked list is implemented.

Functions:
7. Minimum FIVE functions (exclude MAIN function)
8. Minimum FIVE functions should have RETURN type and with minimum ONE parameters but not more than FOUR parameters.

ALL functions must have function prototypes. Fail to do so, 5% will be deducted.

Note:
File processing – your output file or amended file should be organized in such a way that it is easy for user to read when the program is terminated. Disorganized output or amended file – 2% will be deducted.