1 | Experiential Learning Component

1.1 | UNIT II : Experiential Learning - 5

[Level-1: 7Q, Level-3: 1Q]

1. **LEVEL - 1 : Balancing Symbols:** Create a program to check if all types of brackets in a given expression are balanced using a stack.
   - Hint: For every closing bracket, check for its corresponding opening bracket from the stack.
2. **LEVEL - 1 : Infix to Postfix Conversion:** Write a program to convert an infix expression to a postfix expression using a stack.
   - Hint: Use stack precedence to push and pop operators during the conversion process.
3. **LEVEL - 1 : Postfix Expression Evaluation:** Implement a function to evaluate a given postfix expression using a stack.
   - Hint: Use a stack to store operands and apply operations as they appear.
4. **LEVEL - 3 : Infix to Prefix Conversion:** Develop a function to convert an infix expression to a prefix expression.
   - Hint: Consider using two stacks, one for operators and one for operands.
5. **LEVEL - 1 : Convert the following infix expressions to postfix:**
   - $A + B * C$
   - $(A + B) * C$
   - $A + B * C + D$
   - $(A + B) * (C + D)$
6. **LEVEL - 1 : Convert the following infix expressions to prefix:**
   - $A * B + C/D$
   - $(A + B) * (C - D)$
   - $A * (B + C)/D$
   - $A + ((B + C) * D)$
7. **LEVEL - 1 : Convert the following prefix expressions to infix:**
   - $+A * BC$
   - $* + AB - CD$
   - $/A * BCD$
   - $+A * +BCD$
8. **LEVEL - 1 : Convert the following prefix expressions to postfix:**
   - $+A * BCD$
   - $- * A/BCD$
   - $/\ \hat{}ABCD$
   - $+\ \hat{}A * BCD$