

## TEST PLAN

### PROJECT 3

The project is based on implementing a linked - based stack to manipulate infix and postfix expressions. The task is to convert infix expression to postfix, and then evaluate the post-fix expression.

A header file, Stack.h is made in which the linked-based stack is implemented. Class Stack is templated. The following methods are implemented throughout the linked-based stack :-

- push( T value) ;
- pop();
- pop\_back();
- getLength();
- print();
- ~Stack();

In postfix.cpp, the conversion from infix to postfix is done first, and then the postfix expression is further evaluated.

A stack instance of infix, operatorStack, postfix is created.

First, the while loop checks if the input string infixStr is not empty, and then it pushes to the stack.

For evaluating the postfix expression, a stack instance of operandStack is created where the operands are pushed to the stack.

For compiling in gcc, the command `g++ postfix.cpp Stack.h`

To execute , `./a.out`.

**OUTPUT FOR 5 CASES -**

**For postfix evaluation -**

```

Deeptanshus-MacBook-Pro:project3 deeptanshudash$ ./a.out
[Input: (3+4)*(5/2)
34+52/*
[17.5
Deeptanshus-MacBook-Pro:project3 deeptanshudash$ ./a.out
Input: ((6+4))^-34
64+34-^
[0.1
Deeptanshus-MacBook-Pro:project3 deeptanshudash$ ./a.out
Input: (9*9)*9
99*9*
[729

```

```

[Deeptanshus-MacBook-Pro:project3 deeptanshudash$ ./a.out
Input: ((2+3)^5)-1
23+5^1-
3124
[Deeptanshus-MacBook-Pro:project3 deeptanshudash$ ./a.out
Input: 9*(8-7)+(6+5)-4
987-65+4-+*
72
Deeptanshus-MacBook-Pro:project3 deeptanshudash$ █

```

**For Delimiters -**

```

[Deeptanshus-MacBook-Pro:project3 deeptanshudash$ ./a.out
This program checks for properly matched delimiters.
Enter delimited expression (<EOF> to quit) :
(2+3)-(3-9)
Valid
Enter delimited expression (<EOF> to quit) :
((2+2)-
Invalid
Enter delimited expression (<EOF> to quit) :
((2+3)-2)
Valid
Enter delimited expression (<EOF> to quit) :
█

```