COL-216
ASSIGNMENT-2
POSTFIX EXPRESSION CALCULATOR

Made By

Dishant Dhiman *2019CS10347*

R Hari Shankar *2019CS10386*

DESIGN DESCRIPTION

We created a MIPS assembly program to calculate the postfix expression. The postfix expression is taken as input by reading it character by character until we encounter end of line i.e. “\n” (ascii code 10).

We have used the inbuilt $sp stack register for all the stack operation required to evaluate the expression. While pushing the inputs in the stack we keep track of how many numbers are available for evaluation.

If more than 2 numbers are available, then we can perform the stack operation on them successfully and push the final value obtained again to the stack. We continue this process till the end.

For the calculation of *,+,- we pop the top 2 element and evaluate them with the required operator and then push the result back into the stack such that finally only one element will remain in the stack which will be our final answer.

In the end when we check whether the input provided was complete or not, we check whether the numbers available to us are 1 or not. If more than 1 number was available then input is invalid otherwise we pop that number and print it as final answer.

Comments have been added to the code to make it as easily readable and understandable as possible.

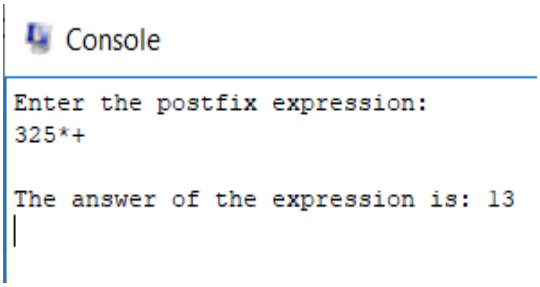


Fig : Example of postfix evaluation

INPUT FORMAT

When the program is run a message will be prompted to input the “Enter the postfix expression” after which we need to input the postfix expression.

FORMAT:

Enter the postfix expression:

<postfix expression>

OUTPUT FORMAT

If the input is of correct format then the output will be displayed else a error message will be displayed stating that the input format is incorrect.

FORMAT:

The answer of expression is: answer

Where answer is the result of the expression

TESTING STRATEGY

To check the correctness of our code to solve the problem of finding the result of postfix expression we have used exhaustive test cases so that our program runs correctly in all cases (including corner cases).

Test Cases:

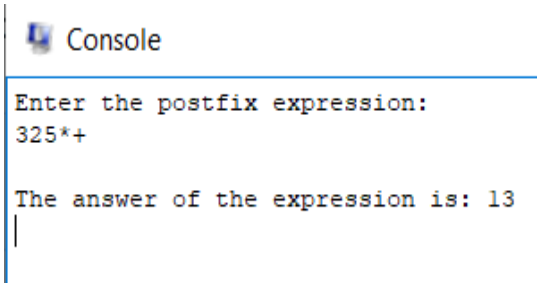
- When the expression is of correct postfix format then correct answer should be displayed.
- Only one number is given as input then that number will be returned as output.

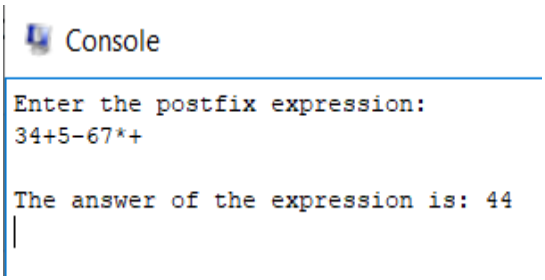
Invalid Input Cases:

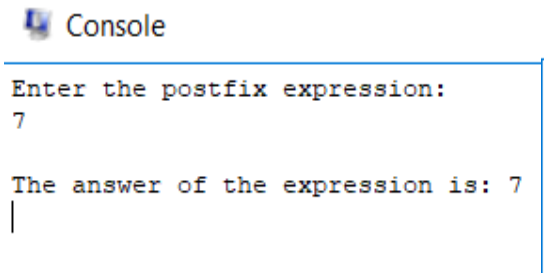
- Two or more numbers without operator.
- One or more operators without numbers.
- Insufficient number of operators.
- More than required operators.
- Nothing is entered in input.
- When operator other than +, *, - is used.

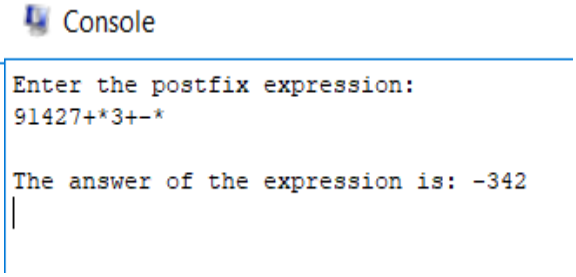
All the testcases used to check correctness of program are given in the file testcases.txt.

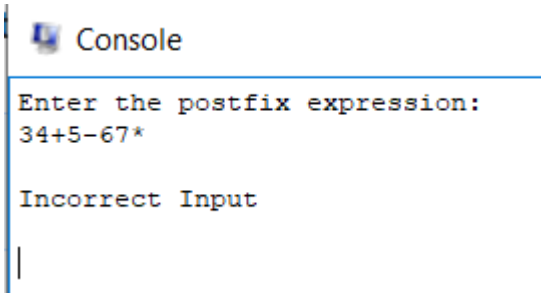
The expected output in the textfile are verified both manually and using online postfix calculator.

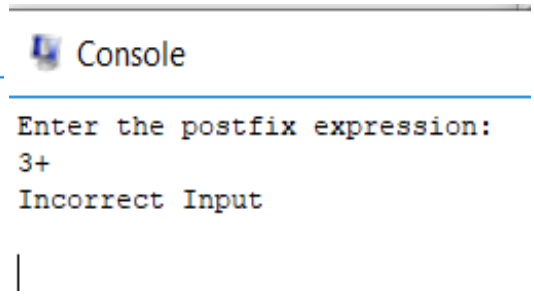

Console
Enter the postfix expression:
325*+
The answer of the expression is: 13

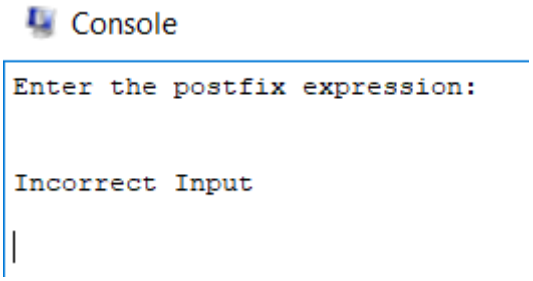

Console
Enter the postfix expression:
34+5-67*+
The answer of the expression is: 44

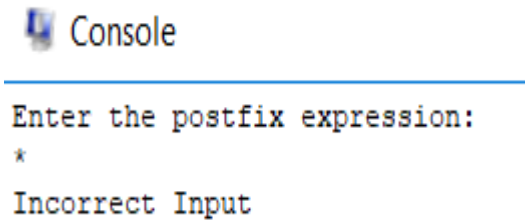

Console
Enter the postfix expression:
7
The answer of the expression is: 7

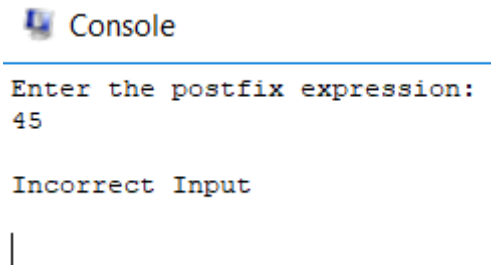

Console
Enter the postfix expression:
91427+*3+-*
The answer of the expression is: -342


Console
Enter the postfix expression:
34+5-67*
Incorrect Input


Console
Enter the postfix expression:
3+
Incorrect Input


Console
Enter the postfix expression:
Incorrect Input


Console
Enter the postfix expression:
*
Incorrect Input


Console
Enter the postfix expression:
45
Incorrect Input