Computer Architecture COL-216
Assignment 2

Aman Gupta,2019CS10673
Arpit Chauhan,2019CS10332
02/03/2021

Description

In this assignment , we have written a MIPS Assembly Program for evaluating an expression given in Postfix format.

Input / Output

- We have been provided with a C++ program which takes an Infix expression as input and gives the corresponding Postfix expression as output . This Postfix expression is then provided as string input to the assembly program at run time .
- As per the required specifications , the assembly program only accepts Postfix expression with constant integer operands in the range 0-9 and operators + , - and *.
- In case the input string contains any other character or the expression itself is not a valid postfix expression , the assembly program throws the appropriate error message on the console .
- For a valid Postfix expression as input , the correct result of the expression is evaluated and consoled out .

Approach / Algorithm

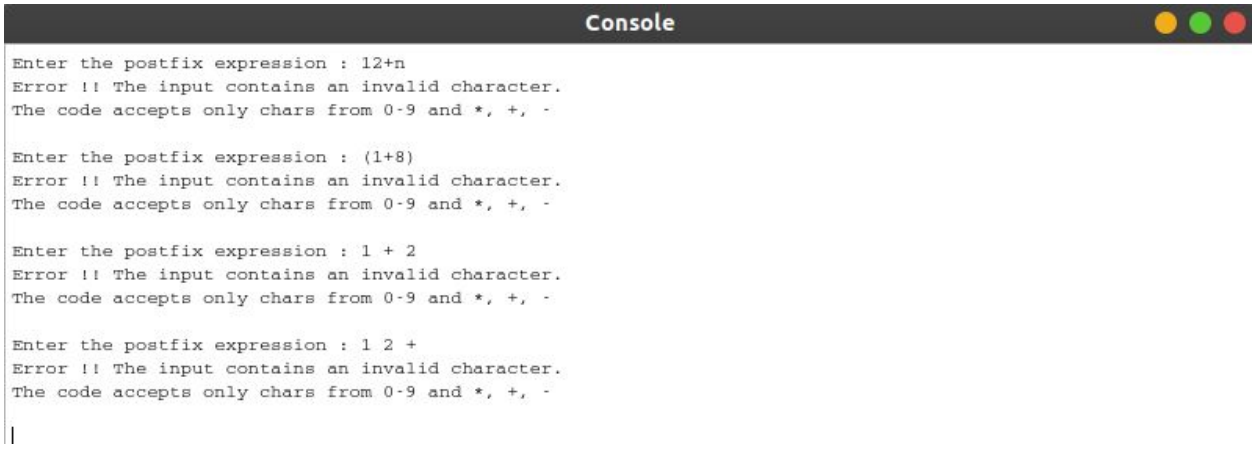
- We have used Stack data structure to evaluate the value of expression given in postfix format .
- We traverse the input string from left to right . Whenever the present character is a valid operand , it is converted into integer and pushed into the stack .
- When we encounter a valid operator in the string , top 2 elements are popped out from the stack , corresponding operation is performed on them and the result is pushed back into the stack . If the stack has less than 2 elements at this step , the postfix expression is invalid and appropriate error is thrown
- If the input expression is in valid Postfix format , only 1 element will be left in the stack after the entire input string has been traversed , which will be the correct result of the expression and the required answer. If the stack has more than 1 element at the end of the algorithm ,the postfix expression is invalid and appropriate error is thrown.

TESTING SYSTEM

Invalid Character Error

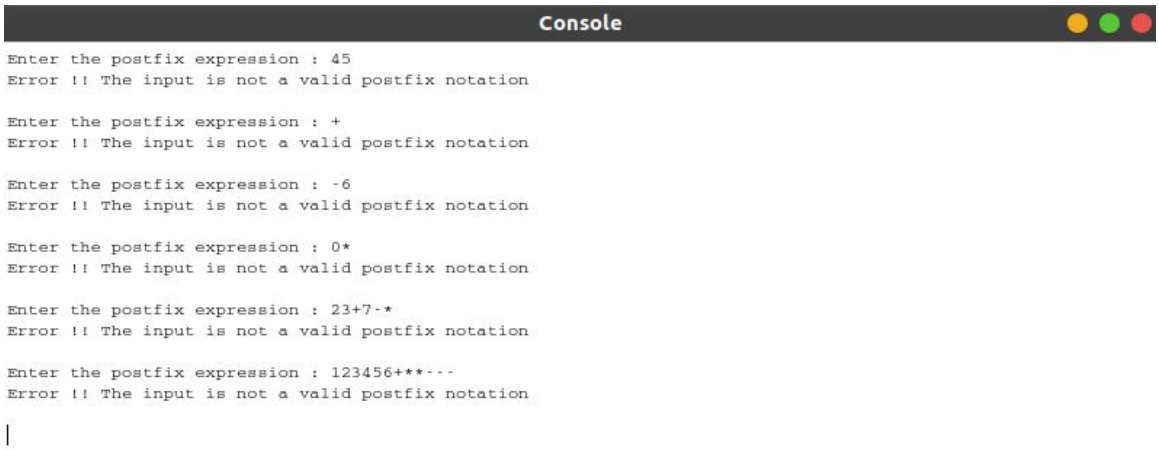
The program only accepts postfix expressions that contain characters from the valid character list.
Valid Character List = {‘0’, ’1’, ’2’, ’3’, ’4’, ’5’, ’6’, ’7’, ’8’, ’9’, ‘+’, ‘-’, ‘*’}

The inputs containing any other character would be rejected by the code with an error message. Even putting extra space in between the Postfix Notation will result in an error.

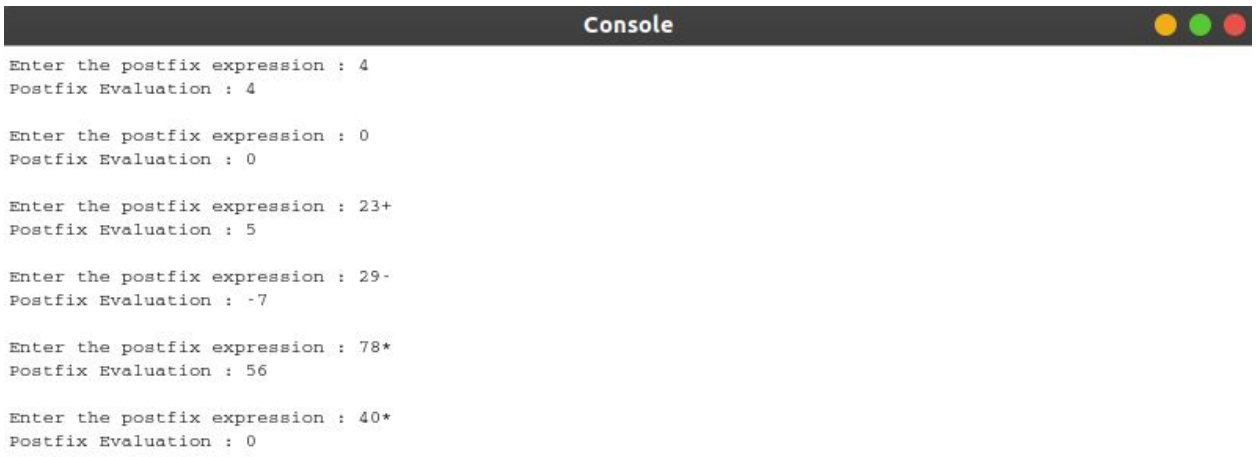


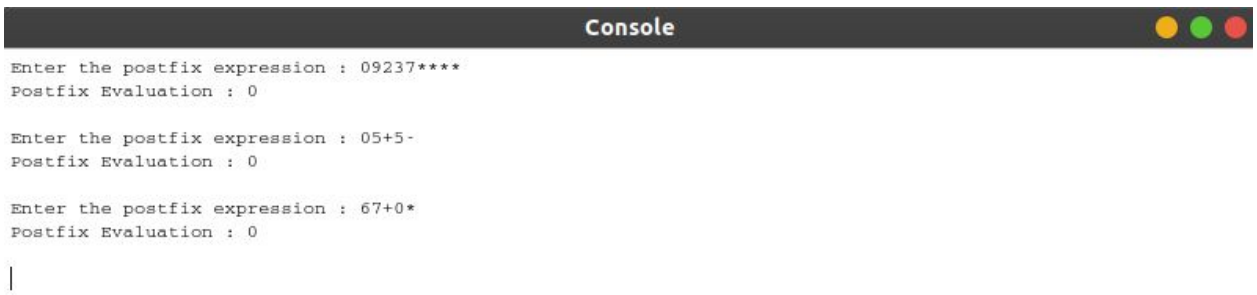
Invalid Postfix Expression Error

If while evaluating the postfix expression the program encounters a stage where there are not enough operands for the operator. It throws an error with the message that the postfix expression is not valid.



Corner and Simple Test Cases



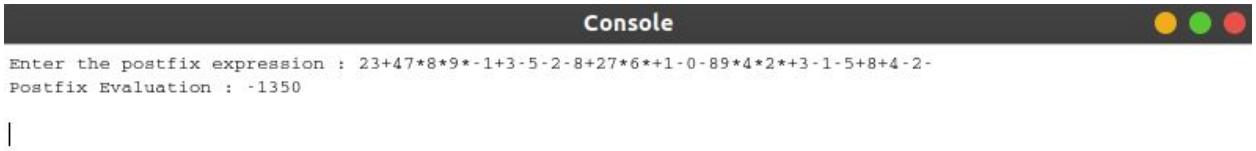


Some large Test Cases

Infix : 2+4-3*6*7-2*2+5-1+3*7-0+9*4-9

Infix : 9*9*9*9*9*9*9

Infix : 2+3-4*7*8*9+1-3-5-2+8+2*7*6-1-0+8*9*4*2-3-1+5+8-4-2



Infix : 3*7*8*6-4-5-2-5-1+8+9+7+5-8*6*4*3*4*9*5-1-2-5+7+8+5+3+7+9*9*6