

Assignment 2: Computer Architecture (COL216)

Divyanka Chaudhari (2019CS50429)

Pranay Gupta (2019CS10383)

March 2, 2021

1 Working of the MIPS code

The console first prints the initial prompt for the user to input the Postfix expression. The input string has been assigned a space of 1028 bytes which means that the user can input a postfix expression of 1026 characters at maximum. Then the input by the user is stored in the register `$a1`. We store the characters '-', '+', '*' in three separate registers which will help in comparison. We have 2 registers `$t3` and `$t4` which store the no. of integers in the postfix expression and no. of operators in the postfix expression respectively that would help in checking validity of postfix expression. We also have a register `$t5` that is initialized to 10 which denotes the end of an input ASCII string.

We then go into the pushLoop branch in which load the first character of the input string into the register `$t0`. We then convert this ASCII character into a integer by subtracting 48 from it. We check if this is one of the operators, else we push the integer into the stack. We increment the address of the string and also increment the counter that checks the number of integers in the stack. If we it is a operator, we go into the add, subtract or multiply loop accordingly. If not one of these, then for a valid postfix expression, it must be the end of an ASCII string character which is 10. If not, then this is an invalid string and we accordingly go into the incorrect loop.

We then go into the addition loop in which the addition of the top 2 elements of the stack is done and the result is pushed into the stack. We increment the counter for the number of operators and then since it is a postfix expression, we check for the next operator and go into adequate branches. Subtract and multiply work similarly.

In the ending loop, we check that the number of integers in the stack must have been one greater than the number of operators, if not then we print "Invalid Postfix Expression". Else, we pop the only element left in the stack which is the result of the computation.

2 Testing

1. **Testcase Description:** Number of operators is greater than or equal to the number of integers

3422345+-**+++

7829++-*

9019++-*

Output: Invalid Postfix Expression

2. **Testcase Description:** Number of operators is less than the number of integers

7980479-+*

78839+*

90332-+*

Output: Invalid Postfix Expression

3. **Testcase Description:** Operator in the middle of integers

32+4234***+

32+5*

433++2*

Output: Invalid Postfix Expression

4. **Testcase Description:** There are multiple zeroes in the input string

09879070-+*+-*+

Output: (-558)

Calculator Output: (-558)

09323010*+-*+-*

Calculator Output: 0

Output: 0

5. **Testcase Description:** Random Test cases

325*+

Output: 13

Calculator Output: 13

3193983+*-+**

Output: (-261)

Calculator Output: (-261)