CSB 353: Compiler Design

LAB 3

Submitted By:

Name: PREM KUMAR

Roll No: 191210037

Branch: CSE

Semester: 6 th

Submitted To: Dr. Shelly Sachdeva

Department of Computer Science and Engineering

NATIONAL INSTITUTE OF TECHNOLOGY DELHI

2019-2023

Ques 1. Write a program using FLEX to read and validate a mathematical expression and display the result. The result should follow the BODMAS Rule. If the expression is invalid then display as invalid

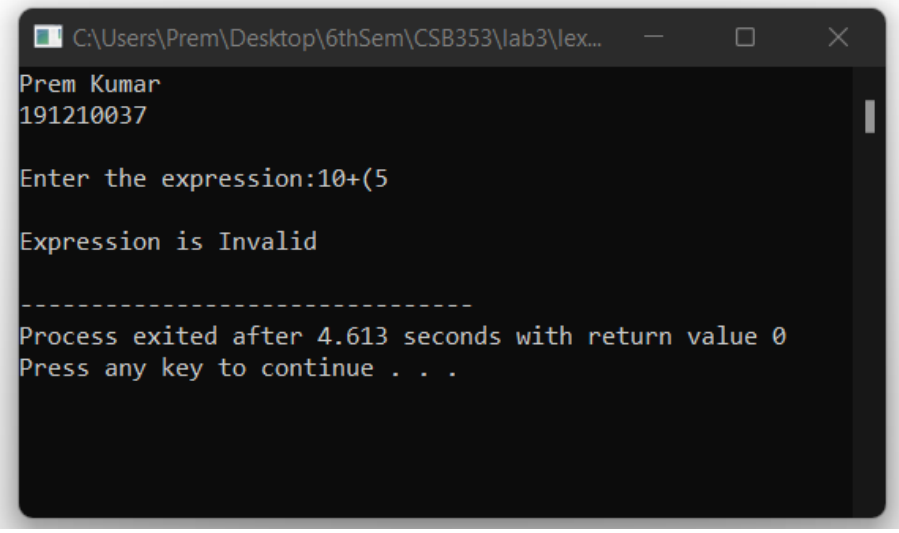
Code:

```
%{
#include<stdio.h>
#include <string.h>
#include <math.h>
int totalOperators = 0, totalOperands = 0, valid = 1, top = -1, eval=0;
char stack[100];
char operator;
int result=0;
%}

%%
"(" {
    top++;
    stack[top] = '(';
}
")" {
    if (stack[top] != '(') {
        valid = 0;
    }
    else if(totalOperands>0 && (totalOperands-totalOperators)!=1){
        valid=0;
    }
    else{
        top--;
        totalOperands=1;
        totalOperators=0;
    }
}
"{" {
    top++;
    stack[top] = '{';
}
"}" {
    if (stack[top] != '{') {
        valid = 0;
    }
    else if(totalOperands>0 && (totalOperands-totalOperators)!=1){
        valid=0;
    }
    else{
        top--;
        totalOperands=1;
        totalOperators=0;
    }
}
```

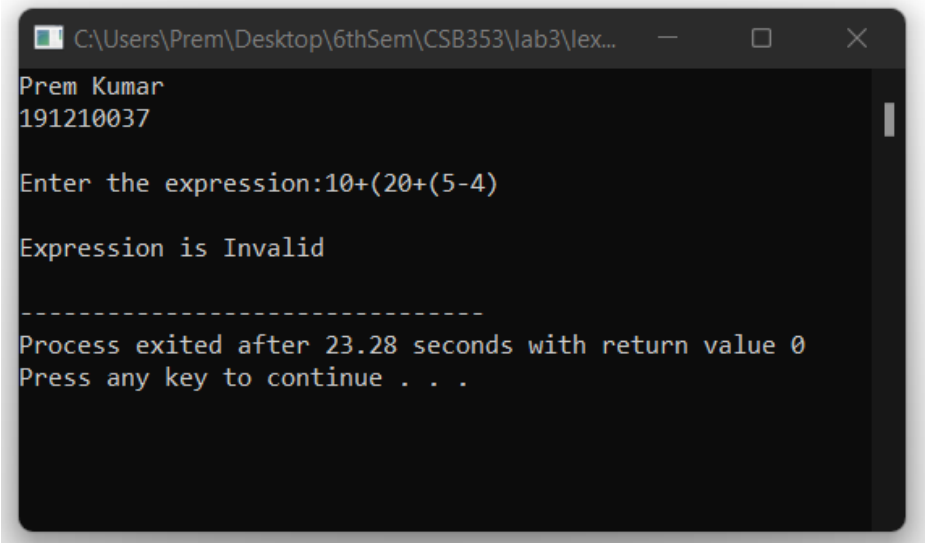
```
"+"|"-"|"*"|"/" {
    totalOperators++;
    operator= yytext[0];
}
[0-9]+|[a-zA-Z][a-zA-Z0-9_]* {
    totalOperands++;
    int n= strlen(yytext);
    int num=0,mul=pow(10,n-1),k=0;
    for(k=0;k<n;k++){
        num+=(yytext[k]-48)*mul;
        mul/=10;
    }
    if(eval==0)
    {   result=num;
        eval=1;
    }else{

        if(operator=='+')
            result+=num;
        else if(operator=='-')
            result-=num;
        else if(operator=='*')
            result*=num;
        else if(operator=='/')
            result/=num;

    }
}
.|\n {return 0;}
%%
```

```
76
77  int main()
78  {
79      printf("Prem Kumar\n191210037\n");
80      printf("\nEnter the expression:");
81      yylex();
82      if(valid==1 && top==-1)
83      {
84              printf("\nExpression is Valid\n");
85              printf("\nResult: %d",result);
86      }
87      else
88              printf("\nExpression is Invalid\n");
89
90      return 0;
91
92  }
93
94  yywrap(void)
95  {
96  return 0;
97  }
```

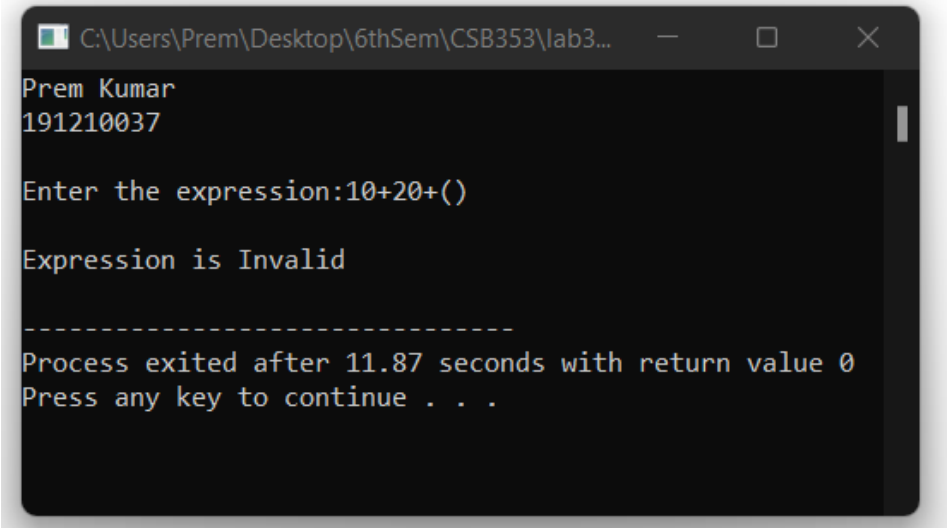
Output:


C:\Users\Prem\Desktop\6thSem\CSB353\lab3\lex...
Prem Kumar
191210037
Enter the expression:10+(5
Expression is Invalid

Process exited after 4.613 seconds with return value 0
Press any key to continue . . .


C:\Users\Prem\Desktop\6thSem\CSB353\lab3\lex...
Prem Kumar
191210037
Enter the expression:10+(20+(5-4)
Expression is Invalid

Process exited after 23.28 seconds with return value 0
Press any key to continue . . .


C:\Users\Prem\Desktop\6thSem\CSB353\lab3...
Prem Kumar
191210037
Enter the expression:10+20+()
Expression is Invalid

Process exited after 11.87 seconds with return value 0
Press any key to continue . . .

Ques 2. Write a program using FLEX to count the number of:

(a) Lines

(b) Words

(c) Capital Letters

(d) Small Letters

(e) Numbers (10,21)

(f) Digits (0-9)

(g) Special Character

(h) Delimiter

(i) Operator

(j) Relational Operator

(k)Total Characters

Code:

```
%{
#include<stdio.h>
int lines=0, words=0,smallLetters=0,capitalLetters=0,digits=0,numbers=0;
int specialCharacters=0,total=0,operators=0,delimiters=0,relationalOperators=0;
%}
%%

[\n] { lines++; words++;}
[\t  ' '] words++;
[A-Z] capitalLetters++;
[a-z] smallLetters++;
[0-9] digits++;
[1-9][0-9]* numbers++;
">"|"<"|"<="|">="|"=="|"!=" {relationalOperators++;operators++;}
"+"|"-"|"*"|"/" operators++;
","|";"|"("|")"|"["|"]"|"{"|"}" delimiters++;
. specialCharacters++;
%%

```

```
20  main(void)
21  {
22  yyin= fopen("input.txt","r");
23  yylex();
24  total=smallLetters+capitalLetters+numbers+digits+specialCharacters+operators+delimiters;
25  printf(" Prem Kumar\n 191210037\n");
26  printf(" The input file contains\n");
27  printf("\n %d lines", lines);
28  printf("\n %d words",words);
29  printf("\n %d small letters", smallLetters);
30  printf("\n %d capital letters",capitalLetters);
31  printf("\n %d digits", digits);
32  printf("\n %d numbers", numbers);
33  printf("\n %d special characters",specialCharacters);
34  printf("\n %d delimiter",delimiters);
35  printf("\n %d operators",operators);
36  printf("\n %d relational Operators",relationalOperators);
37  printf("\n %d Total characters\n",total);
38  }
39
40  int yywrap()
41  {
42  return(1);
43  }
```

Output:

```
C:\Users\Prem\Desktop\6thSem\CSB353\lab3\lex.yy.exe
Prem Kumar
191210037
The input file contains

4 lines
17 words
43 small letters
5 capital letters
2 digits
6 numbers
2 special characters
1 delimiter
2 operators
1 relational Operators
61 Total characters

--------------------------------
Process exited after 0.1143 seconds with return value 22
Press any key to continue . . .
```