Experiment III a

**Aim**: Program to recognize a valid arithmetic expression that uses operators +, -, \* and /.

Algorithm

1. Start
2. Read expression as input
3. Check the expression pattern `p` against regular expressions in lex.
   1. If `p` is an identifier, return ‘id’.
   2. If `p` is a number, return ‘num’.
   3. If `p` is either ‘+’, ‘-’, ‘\*’, ‘/’, ‘(‘, ‘)’ or new line character, return`p`..
4. Check whether the expression satisfies the context free grammar shown below

E -> E/E, E\*E, E+E, E-E, (E)

1. If yes, print “Valid expression”
2. Else print “Invalid expression”
3. Stop

Output

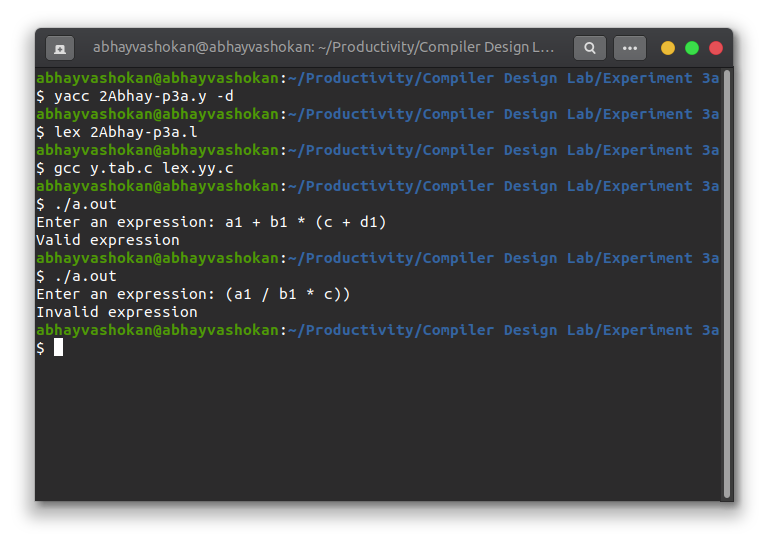
Enter an expression: a1 + b1 \* (c + d1)

Valid expression

Enter an expression: (a1 / b1 \* c))

Invalid expression

Screenshot



Readme

1. Compile the lex program using the command

**lex 2Abhay-P3a.l**

2. Compile the yacc program using the command

**lex 2Abhay-P3a.y -d**

3. Now compile and run the **lex.yy.c** and **y.tab.c** files generated using the command

**gcc lex.yy.c y.tab.c && ./a.out**

4. Input an expression

5. The validated message will be displayed in the terminal.

**Result**: Successfully implemented a program to check if an expression is valid or not.