

PRACTICAL-11

AIM : Create YACC and LEX specification file to recognize arithmetic expression involving +,-,*, /

LEX Program(sample.l)

```
% {
#include<stdio.h>
#include
"y.tab.h"
extern    int
yylval;
% }
%%
[0-9]+ {
        yylval=atoi(y
        ytext); return
        NUMBER;
    }
[\\t] ;
[\\n] return 0;
. return yytext[0];
%%
int yywrap()
{
return 1;
}
```

YACC Program(sample.y)

```
% {
#include<stdio.h>
int
flag=0;

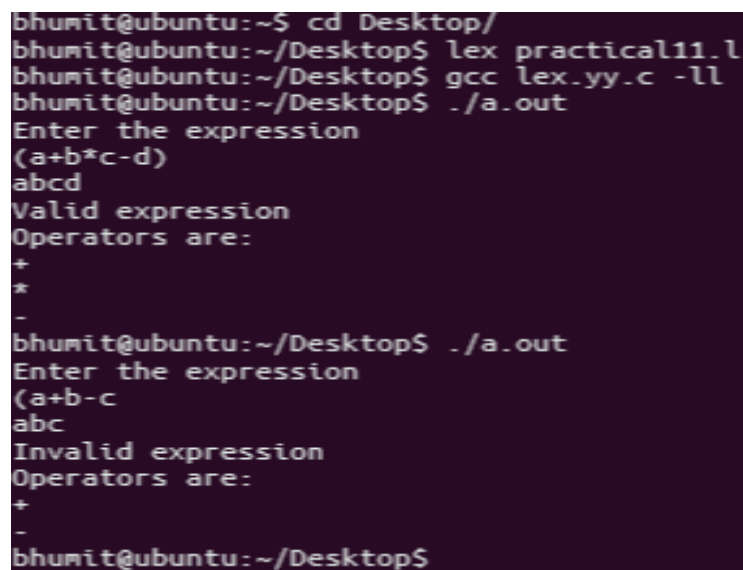
% }
%token NUMBER

%left '+' '-'
%left '*' '/'
%%
ArithmeticExpression: E{
        printf("\\nResult=%d\\n
        ",$$); return 0;
    }
E:E+'E' {$$=$1+$3;}
|E-'E' {$$=$1-$3;}
|E'*E' {$$=$1*$3;}
|E/'E' {$$=$1/$3;}
|NUMBER {$$=$1;}
```

```
;
%%
void main()
{
    printf("\nEnter Any Arithmetic Expression which can have operations Addition,
Subtraction, Multiplication and Divison:\n");
    yypa
rse();
if(fl
g==0
)
    printf("\nEntered arithmetic expression is Valid\n\n");

}
void yyerror()
{
    printf("\nEntered arithmetic expression is
Invalid\n\n"); flag=1;
}
```

OUTPUT :



```
bhunit@ubuntu:~$ cd Desktop/
bhunit@ubuntu:~/Desktop$ lex practical11.l
bhunit@ubuntu:~/Desktop$ gcc lex.yy.c -ll
bhunit@ubuntu:~/Desktop$ ./a.out
Enter the expression
(a+b*c-d)
abcd
Valid expression
Operators are:
+
*
-
bhunit@ubuntu:~/Desktop$ ./a.out
Enter the expression
(a+b-c
abc
Invalid expression
Operators are:
+
-
bhunit@ubuntu:~/Desktop$
```