Name : Nakrani Dhrumil

Roll No : IT082

Lab-9

Calc1.l

```
%{
/* Definition section */
#include<stdio.h>
#include "calc1.tab.h"
extern int yylval;
%}

/* Rule Section */
%%
[0-9]+          {
                    yylval=atoi(yytext);
                    printf("The number formed by LEXER is %s \n ",yytext);
                    return NUMBER;

                    }
[\t]                ;

[\n]                return 0;

['+' | '-']     { printf(" The operator found by LEXER  is %c\n",yytext[0]); return yytext[0];}
.           {printf("\nUNRECOGNIZED SYMBOL FOUND BY LEXER %c \n ",yytext[0]);   return yytext[0];}
```

```
%%

int yywrap()
{
return 1;
}
```

Calc1.y :

```
%{
/* Definition section */
#include<stdio.h>
int flag=0;
%}

%token NUMBER

%left '+' '-'

%left '*' '/' '%'

%left '(' ')'

/* Rule Section */
%%
```

```
ArithmeticExpression: E{

                    /*printf("\nResult=%d\n", $$);*/

                    return 0;

                    };
E: E'+'E { printf("\n Found expression using summation");
          }

 |E'-'E {
          printf("\n Found expression using difference");
         }
 |NUMBER {
           printf ("\n Got number from lexer");
              }
;

%%




//driver code
void main()
```

```
{
printf("\nEnter Any Arithmetic Expression which can have operations Addition and Subtraction:\n");

yyparse();
if(flag==0)
printf("\nEntered arithmetic expression is Valid\n\n");
}

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

Output:

```
C:\Flex Windows\Programs>DDUcompiler

Enter Any Arithmetic Expression which can have operations Addition and  Subtraction:
45+32
The number formed by LEXER is 45

 Got number from lexer The operator found by LEXER  is +
The number formed by LEXER is 32

 Got number from lexer
 Found expression using summation
Entered arithmetic expression is Valid
```