**Ex No: 4 Date:**

# DESIGN A DESK CALCULATOR USING LEX TOOL

# AIM:

To create a calculator that performs addition, subtraction, multiplication and division using lex tool.

# ALGORITHM:

# ● In the headers section declare the variables that is used in the program including

# header files if necessary.

# ● In the definitions section assign symbols to the function/computations we use along

# with REGEX expressions.

# ● In the rules section assign dig() function to the dig variable declared.

# ● In the definition section increment the values accordingly to the arithmetic functions

# respectively.

# ● In the user defined section convert the string into a number using atof() function.

# ● Define switch case for different computations.

# ● Define the main () and yywrap() function.

# PROGRAM:

%{  
int op = 0,i;  
float a, b;  
int digi();  
%}  
  
dig [0-9]+|([0-9]\*)"."([0-9]+)  
add "+"  
sub "-"  
mul "\*"  
div "/"  
pow "^"  
ln \n  
%%  
{dig} {digi();}  
{add} {op=1;}  
{sub} {op=2;}  
{mul} {op=3;}  
{div} {op=4;}  
{pow} {op=5;}  
{ln} {printf("\n The Answer :%f\n\n",a);}  
%%  
int didi();  
int digi()  
{  
if(op==0)  
a=atof(yytext);  
  
else  
{  
b=atof(yytext);  
  
switch(op)  
{  
case 1:a=a+b;  
break;  
  
case 2:a=a-b;  
break;  
  
case 3:a=a\*b;  
break;  
  
case 4:a=a/b;  
break;  
  
case 5:for(i=a;b>1;b--)  
a=a\*i;  
break;  
}  
op=0;  
}  
}  
  
int main(int argv,char \*argc[])  
{  
yylex();  
}  
  
int yywrap()  
{  
return 1;

}

# OUTPUT:

# 

**RESULT:**