

Exp No :

Roll.No: 210701295

Date :

DESIGN A DESK CALCULATOR USING LEX

AIM:

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

ALGORITHM:

Step 1: In the headers section declare the variables that is used in the program including header files if necessary.

Step 2: In the definitions section assign symbols to the function computations we use along with REGEX expressions.

Step 3: In the rules section assign dig() function to the dig variable declared.

Step 4: In the definition section increment the values accordingly to the arithmetic functions respectively.

Step 5: In the user defined section convert the string into a number using atof() function.

Step 6: Define switch case for different computations.

Step 7: Define the main () and yywrap() function.

SOURCE CODE:

```
%{  
int op = 0,i;  
float a, b;  
%}  
  
dig [0-9]+|([0-9]*)."([0-9]+)  
add "+"  
sub "-"  
mul "*"  
div "/"  
pow "^" 1
```

```
{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);}
%%
```

```
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;
```

```
}
```

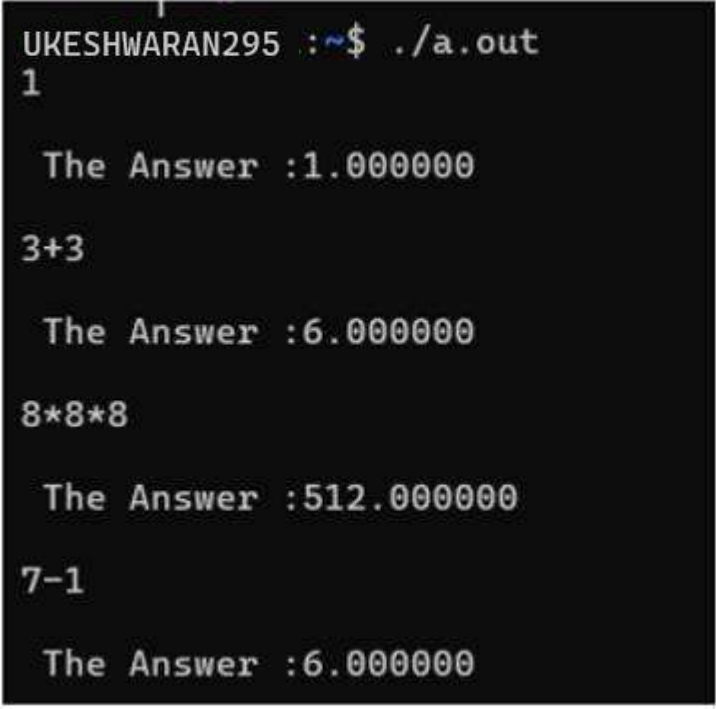
```
}
```

```
main(int argv,char *argc[])
```

```
{
yylex();
}

yywrap()
{
return 1;
}
```

OUTPUT:



A terminal window with a black background and white text. The prompt is 'UKESHWARAN295 :~\$./a.out'. The program outputs the results of four expressions: '1', '3+3', '8*8*8', and '7-1'. Each expression is followed by the text 'The Answer :' and a floating-point result.

```
UKESHWARAN295 :~$ ./a.out
1
The Answer :1.000000
3+3
The Answer :6.000000
8*8*8
The Answer :512.000000
7-1
The Answer :6.000000
```

RESULT: