**CC LAB 08** | Desk Calculator

**Aim:** Lex program to implement a desk calculator with error recovery

**LEX -**

num [0-9]+\.?|[0-9]\*\.[0-9]+

%%

{num} { yylval = (double)atoi(yytext); return num; }

[ ] {}

\n|. { return yytext[0]; }

%%

**YACC –**

%{

#include <stdio.h>

#include <stdlib.h>

#define YYSTYPE double

int yylex(void);

void yyerror(char const\* s);

void push();

%}

%token num

%left '+' '-'

%left '\*' '/'

%right UMINUS

%%

S : S E '\n' {printf("Result = %.2f\n", $2);}

| S '\n'

|

| error '\n' { yyerrok; }

;

E : E'+'E {$$ = $1 + $3;}

| E'-'E {$$ = $1 - $3;}

| E'\*'E {$$ = $1 \* $3;}

| E'/'E {$$ = $1 / $3;}

| '('E')' { $$ = $2; }

| '-' E %prec UMINUS { $$ = -$2; }

| num {$$ = $1;}

;

%%

#include"lex.yy.c"

void yyerror (char const \*s) {

printf("reenter previous line:");

}

int main()

{

printf("Enter calculation to perform: ");

yyparse();

return 0;

}

**Output:**

