**Aim: Implement expression evaluator using LEX and YACC**

P1.l (Lex File)

%{

#include<stdio.h>

#include "y.tab.h"

extern int yylval;

%}

%%

[0-9]+ {yylval=atoi(yytext); return num;}

[-+\*/()] {return yytext[0];}

[ \t]+ {;}

[\n] {return 0;}

%%

P.y (YACC File)

%{

#include<stdio.h>

%}

%token num

%left '+' '-'

%left '/' '\*'

%%

st:exp; {printf("Ans is %d",$$);}

exp:num {$$=$1;}

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

|exp'/'exp {if($3==0){yyerror("Divide By Zero");return 0;}else $$=$1/$3;}

|exp'+'exp {$$=$1+$3;}

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

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

%%

main()

{

yyparse();

}

void yyerror(char \*e)

{

printf("Error : %s",e);

}

To run both files steps are as follows :-

1. Run .y file by command yacc –d p.y

Here -d Cause the header file y.tab.h to be written.

1. Now see y.tab.h file is generated if no error is shown after 1st step

To see file generated or not type ls

1. Now run lex file by lex filename.l command
2. Now compile and link both files by command

gcc lex.yy.c y.tab.c –o vd –ll

Here –o stands for object and –ll for Linking Library

1. Now type object name to run the program.

**OUTPUT:-**

