EX.NO.2

Date:  
  
 **Implementation of a calculator that takes an expression (with**

**digits, + and \*), computes and prints its value, using YACC**

**AIM**

To Implement a calculator that takes an expression (with digits,+ and \*),computes and prints its value, using YACC.

**ALGORITHM**

**STEP 1:**Define the grammar rules for valid arithmetic expressions using YACC, including handling of addition, subtraction, multiplication, division, modulus, and parentheses.

**STEP 2:**Define the tokens for numbers and operators in the LEX file.

**STEP 3:**Include actions in the YACC file to compute the result of the expression and return it.

**STEP 4:**Write a main() function that prompts the user to enter an arithmetic expression and calls the parser to evaluate it.

**STEP 5:**Handle any errors during parsing using the yyerror() function.

**STEP 6:**Compile the LEX and YACC files to generate the executable.

**STEP 7:**Run the calculator program and validate the output with different arithmetic expressions.

**PROGRAM**

%{

#include <stdio.h>

#include <stdlib.h>

int flag = 0;

%}

%token NUMBER

%left '+' '-'

%left '\*' '/' '%'

%left '(' ')'

%%

ArithmeticExpression:

E { printf("\nResult = %d\n", $$); return 0; }

;

E:

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

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

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

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

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

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

| NUMBER { $$ = $1; }

;

%%

int main() {

printf("Enter an expression: ");

yyparse();

if (flag == 0) {

printf("\nEntered arithmetic expression is valid.\n");

}

return 0;

}

void yyerror(const char \*s) {

fprintf(stderr, "Entered arithmetic expression is not valid: %s\n", s);

}  
  
  
  
  
  
%{

#include "pgm\_name.tab.h"

%}

%%

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

[\t] ; // ignore whitespace

\n { return 0; } // end of input

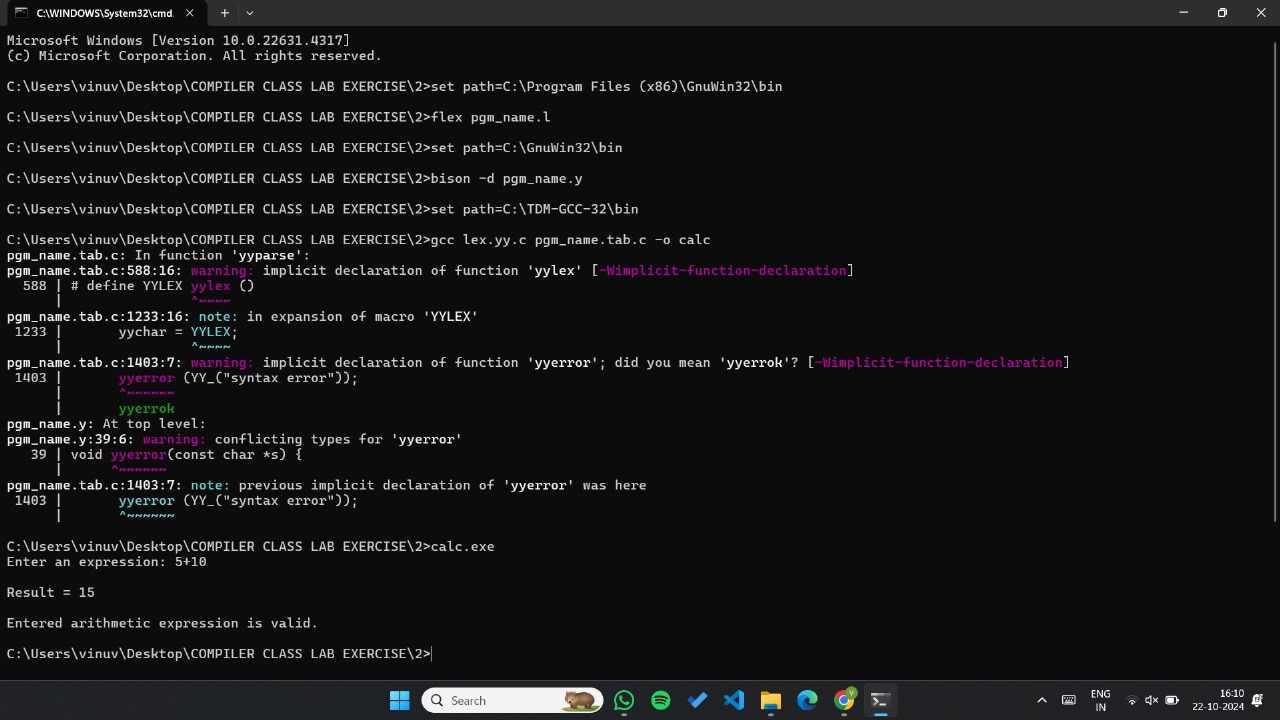
. { return yytext[0]; } // catch all for unrecognized characters

%%

int yywrap() {

return 1;

}

**OUTPUT**

**RESULT**

Thus to Implement lexical analyzer using C program has been executed and verified successfully.