Sharvan Ram Kumaran

Reg No:185001143

CSE C

**Exercise 5**

Objective:

Write Lex program to recognize relevant tokens required for the Yacc parser to implement desk calculator. Write the Grammar for the expression involving the operators namely, + , - ,\* , / , ^ , ( , ).Precedence and associativity has to be preserved. Yacc is available as a command in linux. The grammar should have non terminals E, Op and a terminal id.

**Code:**

Calc.l

%{

#include<stdio.h>

#include<stdlib.h>

#include "y.tab.c"

void yyerror(char \*);

int yylex(void);

int yylval;

%}

GT ">="

LT "<="

AND "&&"

OR "||"

EQ "=="

NEQ"!="

LS "<<"

RS ">>"

%%

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

['+'|'-'|'\*'|'/'|'%'|'^'] {return (\*yytext);}

[\t];

[\n] return 0;

. return yytext[0];

{GT} {return GT;}

{LT} {return LT;}

{AND} {return AND;}

{OR} {return OR;}

{NEQ} {return NEQ;}

{EQ} {return EQ;}

{LS} {return LS;}

{RS} {return RS;}

%%

int yywrap(void)

{

return 1;

}

Calc.y

%{

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

int yylex(void);

void yyerror(char \*);

int flag=0;

%}

%token NUM GT LT RS LS AND EQ NEQ OR

%left OR

%left AND

%left '|'

%left '&'

%left EQ NEQ

%left '>' GT

%left '<' LT

%left LS RS

%left '+' '-'

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

%left '^'

%left '(' ')'

%%

Result: E {printf("Calculated Result=%d",$$);};

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

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

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

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

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

| E '^' E {$$=pow($1,$3);}

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

| E GT E {$$ = $1>= $3; }

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

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

| E LT E {$$ = $1<= $3; }

| E RS E {$$ = $1>>$3; }

| E EQ E {$$ = $1==$3;}

| E NEQ E {$$ = $1!=$3;}

| E LS E {$$ = $1<< $3; }

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

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

| E AND E {$$ = $1 && $3; }

| E OR E {$$ = $1 || $3; }

| NUM {$$ = $1;}

;

%%

int main()

{

printf("\nEnter expression: ");

yyparse();

if(flag==0)

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

return 0;

}

void yyerror(char \* str)

{

printf("\nEntered expression is invalid\n\n");

flag=1;

}

**Output:**

Text

Description automatically generated

Text

Description automatically generated

**Objective:**

A desk calculator was implanted using yacc