/\*Name: ANIKET TRIVEDI

Roll no: B4205

Class: B.E Comp (B)

\*/

/\*Problem Statement:

Implement Scientific Calculator using YACC.

\*/

%{

#include<ctype.h>

#include<stdio.h>

#include<math.h>

#define YYSTYPE double

%}

%token NUMBER

%left '+' '-'

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

%right UMINUS

%left 'S' 'C' 'T' 'L'

%%

lines : lines expr '\n' { printf("%g\n",$2); }

| lines '\n'

| /\* error \*/

;

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

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

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

| expr '/' expr { if($3 == 0)

printf("\n divide by zero error");

else

$$ = $1 / $3;

}

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

| 'S' expr{ $$ = sin($2); }

| 'C' expr{ $$ = cos($2); }

| 'T' expr{ $$ = tan($2); }

| 'L' expr{ $$ = log($2); }

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

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

| NUMBER

;

%%

yylex()

{

int c;

while( (c=getchar()) == ' ' );

if ( (c == '.' ) || ( isdigit (c) ))

{

ungetc(c, stdin);

scanf("%lf",&yylval);

return NUMBER;

}

return c;

}

yyerror()

{

printf("\n ERROR!!!!");

}

main()

{

yyparse();

return 0;

}

/\*

OUTPUT:

comp144@comp144:~$ yaccscalc.y

comp144@comp144:~$ cc y.tab.c -o output -lm

comp144@comp144:~$ ./output

S30

-0.988032

C60

-0.952413

T90

-1.9952

L2

0.693147

2+2

4

2\*20

40

20/2

10

20-10

10

10-20

-10

2/20

0.1

\*/