---------------------------------------------------------------------------------------------------

**cal.y**

%{

#include<stdio.h>

#include<string.h>

#include<math.h>

struct quad

{

char symname[20];

double sval;

};

struct symtab s[20];

int sindex=0,i;

%}

%union

{

char name[20];

double val;

}

%token <name> id

%token <val> num

%token SIN COS TAN SQRT LOG

%left '+' '-'

%left '\*' '/'

%nonassoc UM

%type <val> E

%start SList

%%

SList:SList S

|S;

S:id '=' E ';' {i=insert($1);

s[i].sval=$3;}

|E ';' {printf("%f",$1);}

;

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

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

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

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

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

|SIN'('E')' {$$=sin($3\*(3.141/180.0));}

|COS'('E')' {$$=cos($3\*(3.141/180.0));}

|TAN'('E')' {$$=tan($3\*(3.141/180.0));}

|SQRT'('E')' {$$=sqrt($3);}

|LOG'('E')' {$$=log($3);}

|id {i=insert($1);

$$=s[i].sval;}

|num {$$=$1;}

;

%%

main()

{

yyparse();

}

int insert(char \*name)

{

int i=0;

for(i=0;i<sindex;i++)

{

if(strcmp(s[i].symname,name)==0)

return i;

}

strcpy(s[sindex].symname,name);

sindex++;

return sindex-1;

}

int yyerror(char \*s)

{

printf("%s (Invalid Input)\n",s);

}

**cal.l**

%{

#include "y.tab.h"

%}

%%

"sin" {return SIN;}

"cos" {return COS;}

"tan" {return TAN;}

"sqrt" {return SQRT;}

"log" {return LOG;}

[a-zA-Z\_][a-zA-Z\_0-9]\* {strcpy(yylval.name,yytext); return id;}

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

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

[ \t\n] {;}

"$" {return 0;}

%%

---------------------------------------------------------------------------------------------------

**OUTPUT:**

pccoe@212A-07:~/BE120/assi3$ yacc -d cal.y

pccoe@212A-07:~/BE120/assi3$ lex cal.l

pccoe@212A-07:~/BE120/assi3$ gcc lex.yy.c y.tab.c -ll -lm

pccoe@212A-07:~/BE120/assi3$ ./a.out

a=10;

b=5;

a-b;

5.000000

a=24;

b=6;

a+b;

30.000000

a=100;

b=200;

a\*b;

20000.000000

a=10;

b=2;

a/b;

5.000000

sin(90);

1.000000

cos(90);

0.000296

tan(45);

0.999704

sqrt(169);

13.000000

log(10);

2.302585