

Answer to the question number 1

Lex file:

```
%{  
#include "y.tab.h"  
#include <math.h>  
extern double vbltable[26];  
%}  
%%  
([0-9]+|([0-9]*\.[0-9]+)([eE][+-]?[0-9]+)?) {  
    yylval.dval = atof(yytext); return NUMBER;  
}  
[ \t] ; /* ignore whitespace */  
[a-z] { yylval.vblno = yytext[0] - 'a'; return NAME; }  
(sqrt) {return SQRT;}  
(pow) {return POW;}  
(fact) {return FACT;}  
\\n|. return yytext[0];  
%%
```

Yac file:

```
%{  
#include<stdio.h>
```

STAMFORD UNIVERSITY BANGLADESH
COURSE TITLE: COMPILER LAB
NAME: JOY MOJUMDAR ID: CSE 066 07780

```
#include<stdlib.h>
```

```
#include<math.h>
```

```
extern FILE *yyin;
```

```
double vbltable[26] ;
```

```
void yyerror(const char *c){
```

```
fprintf(stderr,"%s",c);
```

```
}
```

```
int yylex();
```

```
int var1=1;
```

```
int var2=0;
```

```
int i=1;
```

```
%}
```

```
%union{
```

```
double dval;
```

```
int vblno;
```

```
}
```

```
%token <vblno> NAME
```

```
%token <dval> NUMBER
```

```
%token SQRT
```

```
%token POW
```

```
%token FACT
```

```
%type <dval> expression term factor
```

```
%%
```

STAMFORD UNIVERSITY BANGLADESH
 COURSE TITLE: COMPILER LAB
 NAME: JOY MOJUMDAR ID: CSE 066 07780

statement_list: statement_list statement ';' '\n'

| statement ';' '\n'

;

statement: NAME '=' expression

{ vbltable[\$1] = \$3; printf("%c = %lf\n", \$1+'a', \$3); }

| expression { printf("= %g\n", \$1); }

;

expression: expression '+' term { \$\$ = \$1 + \$3; }

| expression '-' term { \$\$ = \$1 - \$3; }

| term { }

;

term: Sqrt '(' term ')' { \$\$ = sqrt(\$3); }

| POW '(' term ',' term ')' { \$\$ = pow(\$3, \$5); }

| FACT '(' term ')' { if(\$3 > 0) {

var2 = \$3;

for(i = 1; i <= var2; i++) {

var1 *= i;

}

\$\$ = var1;

}

}

STAMFORD UNIVERSITY BANGLADESH
COURSE TITLE: COMPILER LAB
NAME: JOY MOJUMDAR ID: CSE 066 07780

```
| factor { }  
  
;  
  
factor: '-' factor { $$ = -$2; }  
  
| '(' expression ')' { $$ = $2; }  
  
| NUMBER { $$ = $1; }
```

```
| NAME { $$ = vbltable[$1]; }
```

```
;
```

```
%%
```

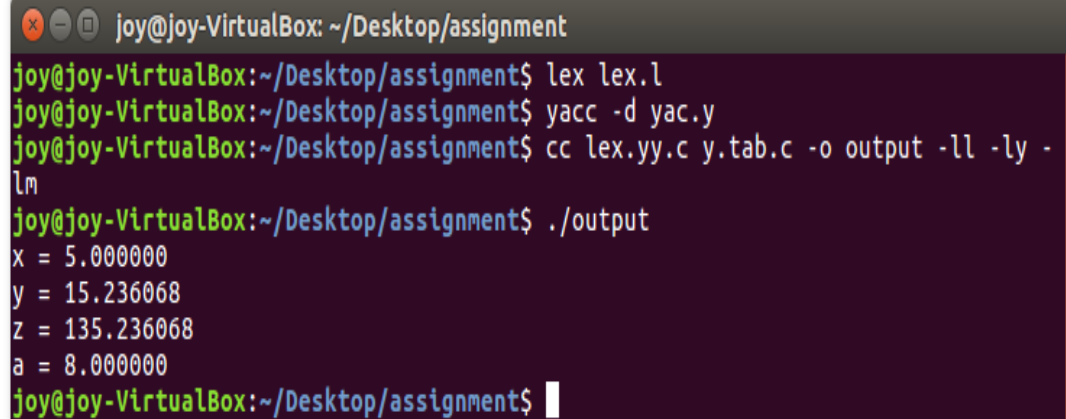
```
int main(){  
FILE *file;  
file = fopen("code.c", "r") ;  
if (!file) {  
printf("Could not open file");  
exit (1);  
}  
else {  
yyin = file;  
}  
yyparse();  
  
}
```

Code.c file:

```
x = 2+3;  
y = sqrt(x)+13;  
z = fact(x)+y;  
a = pow(2,3);
```

output:

```
x = 2+3;  
y = sqrt(x)+13;  
z = fact(x)+y;  
a = pow(2,3);
```



```
joy@joy-VirtualBox: ~/Desktop/assignment  
joy@joy-VirtualBox:~/Desktop/assignment$ lex lex.l  
joy@joy-VirtualBox:~/Desktop/assignment$ yacc -d yac.y  
joy@joy-VirtualBox:~/Desktop/assignment$ cc lex.yy.c y.tab.c -o output -ll -ly -lm  
joy@joy-VirtualBox:~/Desktop/assignment$ ./output  
x = 5.000000  
y = 15.236068  
z = 135.236068  
a = 8.000000  
joy@joy-VirtualBox:~/Desktop/assignment$
```

Answer to the question number 2

Lex file:

```
%{  
#include "y.tab.h"  
extern int yylval;  
%}  
  
%%  
"for" {return FOR;}  
"each" {return EACH;}  
"from" {return FROM;}  
"print" {return PRINT;}  
[ \t]+ {return TAB;}  
'\n' {return NEWLINE;}  
[0-9]+ {yylval = atoi(yytext); return NUMBER;}  
[a-z] {yylval = atoi(yytext - 'a'); return TOKEN;}  
. {return yytext[0];}  
%%
```

Yac file:

```
%{  
#include <stdio.h>  
#include <stdlib.h>  
#include <math.h>  
extern FILE *yyin;  
int yylex();  
int count_start;  
int count_end;  
int var = 0;  
int increment;  
%}  
  
%token FOR EACH FROM PRINT NUMBER TOKEN TAB NEWLINE  
  
%%  
  
function: token token TOKEN token expr ':' expr '{' function2 '}' {count_start = $5;  
count_end = $7; var = $3;};  
  
function2: token token TOKEN '=' TOKEN '+' expr ';' token token statement ';'   
token { increment = $7;}  
  
statement : cmd TOKEN {if(var == $2){
```

STAMFORD UNIVERSITY BANGLADESH

COURSE TITLE: COMPILER LAB

NAME: JOY MOJUMDAR ID: CSE 066 07780

```
for(var = count_start ; var <= count_end; var+=increment){
```

```
    printf("%d ", var);
```

```
}
```

```
}};
```

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

```
    | NUMBER '-' NUMBER { $$ = $1 - $3;}
```

```
    | NUMBER '*' NUMBER { $$ = $1 * $3;}
```

```
    | NUMBER '/' NUMBER { $$ = $1 / $3;}
```

```
    | NUMBER { $$ = $1;}
```

```
cmd : PRINT{};
```

```
token :FROM {}
```

```
    | FOR {}
```

```
    | EACH {}
```

```
    | NEWLINE {}
```

```
    | TAB {}
```

```
;
```

```
%%
```

```
int main(){
```

```
FILE *file;
```

```
file = fopen("code.c", "r");
```

```
if(!file){
```

```
printf("couldn't open file");
```

```
exit(1);
```

```
}
```

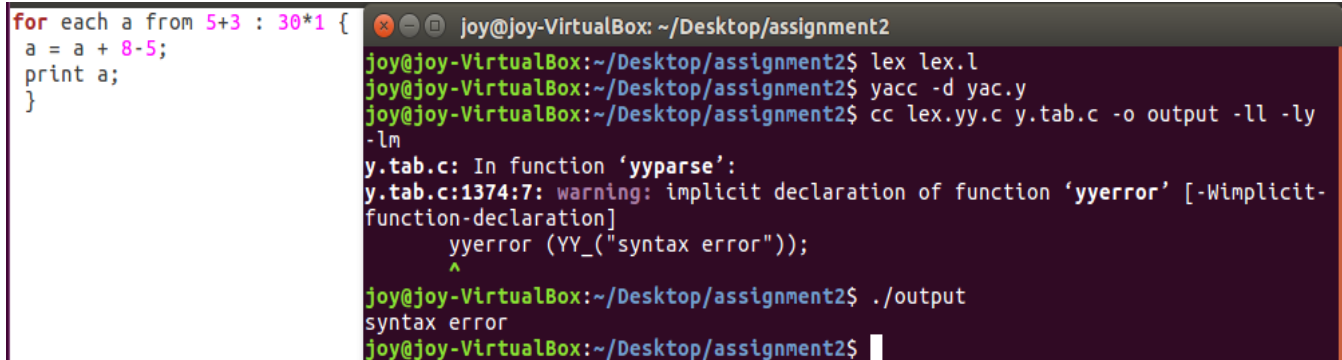

STAMFORD UNIVERSITY BANGLADESH
COURSE TITLE: COMPILER LAB
NAME: JOY MOJUMDAR ID: CSE 066 07780

```
else{  
  
yyin = file;  
  
}  
  
yyparse();  
  
}
```

Code.c file:

```
for each a from 5+3 : 30*1 {  
  
a = a + 8-5;  
print a;  
}
```

Output:



```
for each a from 5+3 : 30*1 {  
a = a + 8-5;  
print a;  
}  
  
joy@joy-VirtualBox: ~/Desktop/assignment2  
joy@joy-VirtualBox:~/Desktop/assignment2$ lex lex.l  
joy@joy-VirtualBox:~/Desktop/assignment2$ yacc -d yac.y  
joy@joy-VirtualBox:~/Desktop/assignment2$ cc lex.yy.c y.tab.c -o output -ll -ly  
-lm  
y.tab.c: In function 'yyparse':  
y.tab.c:1374:7: warning: implicit declaration of function 'yyerror' [-Wimplicit-  
function-declaration]  
        yyerror (YY_("syntax error"));  
        ^  
joy@joy-VirtualBox:~/Desktop/assignment2$ ./output  
syntax error  
joy@joy-VirtualBox:~/Desktop/assignment2$
```