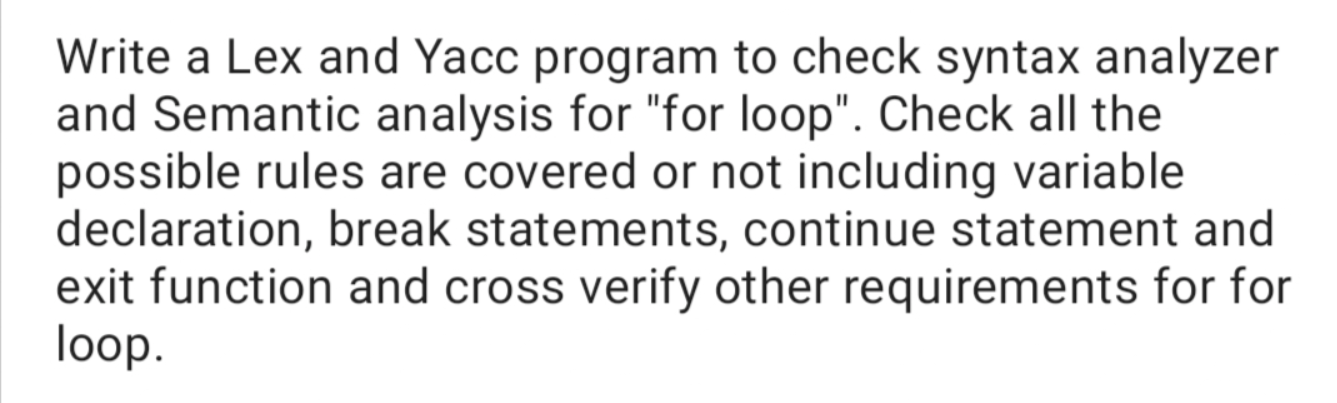
**LEX ASSIGNMENT – 6**

**BY:K.S.ABHIHKESHAV**

**21CSB0F03**



**Lex:**

%{

#include "y.tab.h"

%}

DIGIT [0-9]

LETTER [a-zA-Z]

IDENTIFIER ({LETTER}|\_)({LETTER}|{DIGIT}|\_)\*

QUOTE ["]

%%

"{" { return LBRACE; }

"}" { return RBRACE; }

"(" { return LPAREN; }

")" { return RPAREN; }

";" { return SEMICOLON; }

if { return IF; }

else { return ELSE; }

for { return FOR; }

continue {return CONTINUE;}

break {return BREAK;}

printf {return PRINT;}

int { return INT; }

float { return FLOAT; }

char { return CHAR; }

return { return RETURN; }

{IDENTIFIER} { return IDENTIFIER; }

{DIGIT}+ { return NUMBER; }

"++"|"--" {return SINGLEOP; }

"+="|"-="|"\*="|"/=" {return SELFOP; }

">"|"<"|"=="|">="|"<=" {return COMPARATOR; }

"+"|"-"|"/"|"\*" {return OPERATIONS; }

"=" {return ASSIGN; }

[ \t\n]+ /\* ignore whitespace \*/

{QUOTE}.+{QUOTE} {return STRING;}

. {return yytext[0];}

%%

int yywrap() {

return 1;

}

**YACC:**

%{

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int flag=0;

extern FILE \*yyin;

%}

%token IDENTIFIER NUMBER COMPARATOR OPERATIONS SINGLEOP ASSIGN SELFOP

%token IF ELSE FOR CONTINUE BREAK PRINT STRING

%token INT FLOAT CHAR RETURN

%token LBRACE RBRACE LPAREN RPAREN SEMICOLON

%start forstart

%%

forstart:

FOR LPAREN declar\_statement SEMICOLON comparison SEMICOLON op\_expression RPAREN LBRACE statement\_list RBRACE

;

statement\_list:

/\* empty \*/

| statement\_list statement

;

statement:

expression SEMICOLON

| selection\_statement

| declar\_statement SEMICOLON

| forstart

| CONTINUE SEMICOLON

| BREAK SEMICOLON

| ret\_statement SEMICOLON

| print\_statement SEMICOLON

;

print\_statement:

PRINT LPAREN STRING RPAREN

;

declar\_statement:

data\_type IDENTIFIER

| data\_type IDENTIFIER ASSIGN operation

;

ret\_statement:

RETURN IDENTIFIER

| RETURN NUMBER

| RETURN

;

expression:

op\_expression

| comparison

;

op\_expression:

IDENTIFIER ASSIGN operation

| IDENTIFIER SELFOP operation

| operation

;

comparison:

either COMPARATOR either

;

operation:

either OPERATIONS operation

| IDENTIFIER SINGLEOP

| either

;

either:

NUMBER

| IDENTIFIER

;

data\_type:

INT | FLOAT | CHAR

;

selection\_statement:

IF LPAREN expression RPAREN LBRACE statement\_list RBRACE

| IF LPAREN expression RPAREN LBRACE statement\_list RBRACE ELSE LBRACE statement\_list RBRACE

;

%%

int main() {

yyin=fopen("input.txt","r");

yyparse();

if(flag==0)

printf("Accepted\n");

return 0;

}

void yyerror(const char\* message) {

flag=1;

printf("Rejected\n");

}

**Input and output:**

