Compiler Construction

Practical 7

19BCE248

DL2

AIM: To implement grammar rules for control statements, and Loop control.

Yacc file:

%{

#include<stdio.h>

#include <string.h>

%}

%token IF ELSE KEYWORD UN FOR ID NUM OPEN CLOSE GT LT EQ NTE GTE LTE AS SC OPENP CLOSEP WHILE OP

%%

START: OPEN ST CLOSE {printf("Done"); exit(0);}

ST : EXP STS | FORST STS | WHILEST STS | IFST STS | ELSE STS | ;

STS: ST | OPEN ST CLOSE ST | SC STS ;

IFST: IF OPENP REL CLOSEP ;

REL: Z1;

WHILEST : WHILE OPENP REL CLOSEP ;

FORST: FOR OPENP EXP1 EXP2 EXP3 CLOSEP ;

EXP1: EXP | Z1 | SC | Z R Z SC;

EXP2 : Z1 SC | ID AS Z OP Z SC ;

EXP3: ID UN | Z | ID AS Z | ;

EXP: ID UN SC| ID AS Z OP Z1 SC | ID AS Z1 SC| KEYWORD ID AS Z1 SC| KEYWORD ID AS Z OP Z1 SC| KEYWORD ID SC| Z SC ;

Z1: Z| Z R Z;

Z:ID | NUM ;

R: GT | LT | EQ | NTE | GTE | LTE;

%%

int yyerror(){

printf("some error");

return 0;

}

int main(){

yyparse();

return 0;

}

Lex file:

%{

#include "prac7.tab.h"

%}

alpha [A-Za-z]

digit [0-9]

%%

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

"int"|"float"|"char"|"double"|"struct"|"do"|"printf"|"return" {return KEYWORD;}

"while" {return WHILE;}

("if") {return IF;}

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

("else") {return ELSE;}

("for") {return FOR;}

{digit}+ {return NUM;}

{alpha}({alpha}|{digit})\* {return ID;}

("{") {return OPEN;}

("}") {return CLOSE;}

(">") {return GT;}

("<") {return LT;}

(">=") {return GTE;}

("<=") {return LTE;}

("!=") {return NTE;}

("==") {return EQ;}

("=") {return AS;}

(";") {return SC;}

("(") {return OPENP;}

(")") {return CLOSEP;}

("//").\* {}

%%

int yywrap(){

return 0;

}

Test file :

{

for(int i=0;i<n;i++){

while(j<m){

if(x==10){

b++;

}else{

z++;

}

}

if(a==b){

while(a>0){

a--;

}

}

if(z==10)

for(int i=0;i<n;i++)

b++;

}

int a = 0;

if ( a < b )

{

a = b;

}

else

{

b = a;

}

for(;1;) ;

for(1;2;3);

for(int i=b+c<d;i<5;i++);

for(int i;i<5;i++);

//for(a<b;int i=b+c<d;i<5);

for(a<b;a=a+1; a=0)

for(int i=0;i<5;i++)

{

y = 0;

x=y+c;

}

int i = 0;

while ( i < 3)

{

y = y + 5;

}

}

Output:



