P2.l

%{

#include "y.tab.h"

%}

%%

"if" { return IF; }

[sS][0-9]\* { return S; }

"<"|">"|"=="|"!="|"<="|">=" { return RELOP; }

[0-9]+ { return NUMBER; }

[a-zA-Z][a-zA-Z0-9\_]\* { return ID; }

\n { /\* ignore newlines \*/ }

[ \t] { /\* ignore whitespace \*/ }

. { return yytext[0]; }

%%

int yywrap(void) {

return 1;

}

P2.y

%{

#include <stdio.h>

#include <stdlib.h>

int count = 0;

// Declare yylex and yyerror to avoid implicit declaration warnings

int yylex(void);

void yyerror(const char \*msg);

%}

%token IF RELOP S NUMBER ID

%%

stmt : if\_stmt { printf("No of nested if statements = %d\n", count); exit(0); }

;

if\_stmt : IF '(' cond ')' if\_stmt { count++; }

| S

;

cond : x RELOP x

;

x : ID

| NUMBER

;

%%

void yyerror(const char \*msg) {

printf("Invalid Expression\n");

exit(0);

}

int main() {

printf("Enter the statement: ");

yyparse();

return 0;

}

Steps:

lex P2.l

Yacc -d P2.y

gcc lex.yy.c y.tab.c

./a.out

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

Output:

Enter the statement: if (x > 3) if (y == 2) s

No of nested if statements = 2