# **Laboratory-3**

## **Question-1**

## Write a code for syntax analysis of if and if-else conditional constructs in C.

### **Lex File (lex.l):**

%{

#include "y.tab.h"

%}

%%

"if" { **return** IF; }

"else" {**return** ELSE; }

"(" { **return** LPAREN; }

")" { **return** RPAREN; }

[a-zA-Z][a-zA-Z0-9]\* { printf("%s\n",yytext);**return** ID; }

. ;

%%

**int** yywrap() {

**return** 1;

}

### **Yacc File (par.y):**

%{

#include <stdio.h>

#include <stdlib.h>

**extern** **int** yylex();

**extern** **int** yyparse();

**extern** FILE\* yyin;

**void** yyerror(**const** **char**\*);

#define YYSTYPE YYSTYPE

#define YYDEBUG 1

**typedef** **struct** {

**char**\* str;

} YYSTYPE;

%}

%token IF LPAREN RPAREN ELSE ID

%%

program : statement;

statement : if\_statement

| ID

;

if\_statement : IF LPAREN ID RPAREN statement

| IF LPAREN ID RPAREN statement ELSE statement

;

%%

**void** yyerror(**const** **char** \*s) {

printf("Error!\n");

exit(0);

}

**int** main() {

FILE\* input\_file = fopen("sample.c", "r");

yyin = input\_file;

yyparse();

printf("success\n");

fclose(input\_file);

**return** 0;

}

### **Sample File (sample.c):**

**if** (askdfjalksfj){

fihaskjf

}

**else**{

sofjalsjf

}

### Output:

A white background with blue text

Description automatically generated

## **Result:**

Code for syntax analysis of conditional constructs was implemented successfully.

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

## **Question-2**

## Write code for syntax analysis of while loop constructs in C.

### **Lex File (lex.l):**

%{

#include "y.tab.h"

%}

%%

"while" { **return** WHILE; }

"(" { **return** LPAREN; }

")" { **return** RPAREN; }

[a-zA-Z][a-zA-Z0-9]\* { printf("%s\n",yytext);**return** ID; }

. ;

%%

**int** yywrap() {

**return** 1;}

### **Yacc File (par.y):**

%{

#include <stdio.h>

#include <stdlib.h>

**extern** **int** yylex();

**extern** **int** yyparse();

**extern** FILE\* yyin;

**void** yyerror(**const** **char**\*);

#define YYSTYPE YYSTYPE

#define YYDEBUG 1

**typedef** **struct** {

**char**\* str;

} YYSTYPE;

%}

%token WHILE LPAREN RPAREN ID

%%

statement : WHILE LPAREN ID RPAREN statement

| ID

;

%%

**void** yyerror(**const** **char** \*s) {

printf("Error!\n");

exit(0);

}

**int** main() {

FILE\* input\_file = fopen("sample.c", "r");

yyin = input\_file;

yyparse();

printf("success\n");

fclose(input\_file);

**return** 0;

}

### **Sample File (sample.c):**

**while**(dgs){

dgdshs

}

### **Output:**

A white background with blue text

Description automatically generated

**Result:**

Code for syntax analysis of loop constructs was implemented successfully.