**CC Lab Assignment 10**

**Aim:** Write a note on Intermediate Code with an example. What is its significance?

What is meant by three address code? What are the ways of representing three address code?

Write a program in LEX and YACC to implement Intermediate code (IC) generator for arithmetic expression.

**Code:**

**Lex File:**

%{

#include<stdio.h>

#include<string.h>

#include "y.tab.h"

%}

%%

[0-9]+ {strcpy(yylval.n, yytext); return oprnd;}

[a-zA-Z\_][a-zA-Z0-9\_]\* {strcpy(yylval.n, yytext); return oprnd;}

"+" {return PLUS;}

"\*" {return MUL;}

"=" {return '=';}

[()] {return yytext[0];}

[ \t\n\r]+ {}

. {}

%%

int yywrap()

{

return 1;

}

**Yacc File:**

%{

#include<stdio.h>

#include<string.h>

#include "y.tab.h"

void yyerror(const char \*s);

int yylex();

int temp = 0;

void generate\_code(char op1[], char op[], char op2[], char res[]) {

sprintf(res, "t%d", temp++);

printf("%s = %s %s %s\n", res, op1, op, op2);

}

%}

%union {

char n[10];

}

%token <n> oprnd;

%token PLUS MUL '='

%type <n> I E T F

%%

I: oprnd '=' E {

printf("%s = %s\n", $1, $3);

printf("-----\n");

};

E: E PLUS T {

char tmp[10];

generate\_code($1, "+", $3, tmp);

strcpy($$, tmp);

}

| T { strcpy($$, $1); }

;

T: T MUL F {

char tmp[10];

generate\_code($1, "\*", $3, tmp);

strcpy($$, tmp);

}

| F { strcpy($$, $1); }

;

F: oprnd { strcpy($$, $1); }

| '(' E ')' { strcpy($$, $2); }

;

%%

int main() {

printf("Enter an expression\n");

yyparse();

return 1;

}

void yyerror(const char \*s) {

fprintf(stderr, "Error: %s\n", s);

}

Output:

