**Name: Aniket Dongre**

**Roll no: 17**

**Div: E GR: 1710446**

**Batch: B1**

**PROBLEM: Code Optimization**

**PROGRAM:**

// LEX

%{

#include"y.tab.h"

extern char yyval;

%}

%%

[0-9]+ {yylval.symbol = (char)yytext[0];return NUMBER;}

[a-zA-Z]+ {yylval.symbol =(char) yytext[0];return ID;}

. {return yytext[0];}

\n {return 0;}

%%

//YACC

%{

#include"y.tab.h"

#include<stdio.h>

#include<stdlib.h>

char temp ='A'-1;

int index1=0;

char addtotable(char, char, char);

struct expr{

char operand1;

char operand2;

char operator;

char result;

};

%}

%union{

char symbol;

}

%left '+''-'

%left '\*''/'

%token <symbol> NUMBER ID

%type <symbol> exp

%%

st: ID '=' exp ';' {addtotable((char)$1,(char)$3,'=');};

exp: exp '+' exp {$$ = addtotable((char)$1,(char)$3,'+');}

|exp '-' exp {$$ = addtotable((char)$1,(char)$3,'-');}

|exp '/' exp {$$ = addtotable((char)$1,(char)$3,'/');}

|exp '\*' exp {$$ = addtotable((char)$1,(char)$3,'\*');}

|'(' exp ')' {$$ = (char)$2;}

|NUMBER {$$ = (char)$1;}

|ID {$$=(char)$1;};

%%

struct expr arr[20];

void quad(){

int i;

for(i=0;i<index1;i++){

if(arr[i].operator=='!') continue;

printf("%c:=\t",arr[i].result);

printf("%c\t",arr[i].operand1);

printf("%c\t",arr[i].operand2);

printf("%c\n",arr[i].operator);

}

}

int main(){

temp='A'-1;

printf("Enter the expression\n");

yyparse();

quad();

opt();

printf("After Optimization\n");

quad();

}

int yywrap(){

return 1;

}

void yyerror(char \*s){

printf("Error %s",s);

}

char addtotable(char a, char b, char c){

temp++;

arr[index1].operand1=a;

arr[index1].operand2=b;

arr[index1].operator=c;

arr[index1].result=temp;

index1++;

return temp;

}

void opt(){

int i,j;

for(i=0;i<index1;i++)

for(int j=i+1;j<index1;j++){

if(arr[i].operator==arr[j].operator && arr[i].operand1 ==arr[j].operand1 && arr[i].operand2 == arr[j].operand2){

int z;

for(int z=j+1;z<index1;z++){

if(arr[z].operand1==arr[j].result) arr[z].operand1=arr[i].result;

if(arr[z].operand2==arr[j].result) arr[z].operand2=arr[i].result;

}

arr[j].operator='!';

}

}

}



