**Practical 10**

**Roll No:** 18BCE194

**Name:** Jay Rathod

**Aim:** To implement code optimization technique.

**Code:**

#include<stdio.h>

#include<conio.h>

#include<string.h>

#include <bits/stdc++.h>

using namespace std;

struct op

{

char l;

char r[20];

}op[10],pr[10];

int main()

{

int a,i,k,j,n,z=0,m,q;

char \*p,\*l;

char temp,t;

char \*tem;

cout<<"Enter the Number of Values:"<<endl;

cin>>n;

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

{

cout<<"left: ";

op[i].l=getche();

cout<<"\tright: ";

scanf("%s",op[i].r);

}

cout<<"\nIntermediate Code"<<endl;

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

{

cout<<op[i].l<<op[i].r<<endl;

}

for(i=0;i<n-1;i++)

{

temp=op[i].l;

for(j=0;j<n;j++)

{

p=strchr(op[j].r,temp);

if(p)

{

pr[z].l=op[i].l;

strcpy(pr[z].r,op[i].r);

z++;

}

}

}

pr[z].l=op[n-1].l;

strcpy(pr[z].r,op[n-1].r);

z++;

cout<<"After Dead Code Elimination"<<endl;

for(k=0;k<z;k++) {

cout<<pr[k].l<<"\t="<<pr[k].r<<endl;

}

for(m=0;m<z;m++) {

tem=pr[m].r;

for(j=m+1;j<z;j++)

{

p=strstr(tem,pr[j].r);

if(p)

{

t=pr[j].l;

pr[j].l=pr[m].l;

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

{

l=strchr(pr[i].r,t) ;

if(l)

{

a=l-pr[i].r;

cout<<"pos: "<<a<<endl;

pr[i].r[a]=pr[m].l;

}

}

}

}

}

cout<<"Eliminate Common Expression"<<endl;

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

cout<<pr[i].l<<"\t="<<pr[i].r<<endl;

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

{

for(j=i+1;j<z;j++)

{

q=strcmp(pr[i].r,pr[j].r);

if((pr[i].l==pr[j].l)&&!q)

{

pr[i].l='\0';

strcpy(pr[i].r,'\0');

}

}

}

cout<<"Optimized Code"<<endl;

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

{

if(pr[i].l!='\0')

cout<<pr[i].l<<"="<<pr[i].r<<endl;

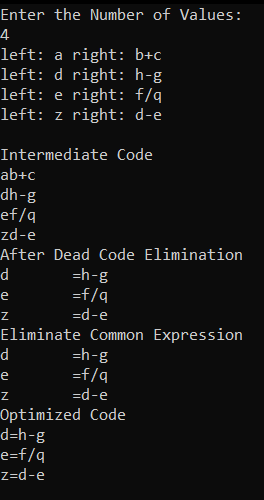
}

getch();

return 0;

}

**Output:**



**Conclusion:**

Learned about code optimization technique and implemented the same using cpp.