**Experiment No. 8**

**Aim: To implement Code Optimization techniques**

Code:

#include<stdio.h>

#include<string.h>

#include<ctype.h>

void input();

void output();

void change(int p,int q,char\*res);

void expression();

void constant();

struct expr

{

char op[2],op1[5],op2[5],res[5];

int flag;

}

arr[10];

int n;

void main()

{

input();

constant();

expression();

output();

getch();

}

void input()

{

int i;

printf("\n\nEnter the maximum number of ecpressions:");

scanf("%d",&n);

printf("\n Enter the input:\n");

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

{

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

scanf("%s",arr[i].op1);

scanf("%s",arr[i].op2);

scanf("%s",arr[i].res);

arr[i].flag=0;

}

}

void constant()

{

int i;

int op1,op2,res;

char op,res1[5];

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

{

if(isdigit(arr[i].op1[0])&&isdigit(arr[i].op2[0]))//if both digits,store them in variables

{

op1=atoi(arr[i].op1);

op2=atoi(arr[i].op2);

op=arr[i].op[0];

switch(op)

{

case'+':

res=op1+op2;

break;

case'-':

res=op1-op2;

break;

case'\*':

res=op1\*op2;

break;

case'/':

res=op1/op2;

break;

}

sprintf(res1,"%d",res);

arr[i].flag=1;//eliminate expr and replace any operand below that uses result of this expr

change(i,i,res1);

}

}

}

void expression()

{

int i,j;

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

{

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

{

if(strcmp(arr[i].op,arr[j].op)==0)

{

if(strcmp(arr[i].op,"+")==0||strcmp(arr[i].op,"\*")==0)

{

if(strcmp(arr[i].op1,arr[j].op1)==0&&strcmp(arr[i].op2,arr[j].op2)==0||strcmp(arr[i].op1,arr[j].op2)==0&&strcmp(arr[i].op2,arr[j].op1)==0)

{

arr[j].flag=1;

change(i,j,NULL);

}

}

else

{

if(strcmp(arr[i].op1,arr[j].op1)==0&&strcmp(arr[i].op2,arr[j].op2)==0)

{

arr[j].flag=1;

change(i,j,NULL);

}

}

}

}

}

}

void output()

{

int i=0;

printf("\nOptimized code is:");

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

{

if(!arr[i].flag)

{

printf("\n %s %s %s %s",arr[i].op,arr[i].op1,arr[i].op2,arr[i].res);

}

}

}

void change(int p,int q,char\*res)

{

int i;

for(i=q+1;i<n;i++)

{

if(strcmp(arr[q].res,arr[i].op1)==0)

if(res==NULL) //for csub

strcpy(arr[i].op1,arr[p].res);

else //for ceval

strcpy(arr[i].op1,res);

else if(strcmp(arr[q].res,arr[i].op2)==0)

if(res==NULL) //for csub

strcpy(arr[i].op2,arr[p].res);

else //for ceval

strcpy(arr[i].op2,res);

}

}

**Output:**

