**3a. Design and implement C/C++ Program to solve All-Pairs Shortest Paths problem using Floyd's algorithm.**

#include<stdio.h>

#define INF 999

int min(int a,int b)

{

return(a<b)?a:b;

}

void floyd(int p[][10],int n)

{

int i,j,k;

for(k=1; k<=n; k++)

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

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

p[i][j]=min(p[i][j],p[i][k]+p[k][j]);

}

void main()

{

int a[10][10],n,i,j;

printf("\nEnter the n value:");

scanf("%d",&n);

printf("\nEnter the graph data:\n");

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

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

scanf("%d",&a[i][j]);

floyd(a,n);

printf("\nShortest path matrix\n");

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

{

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

printf("%d ",a[i][j]);

printf("\n");

}

}

**OUTPUT**

Enter the n value:4

Enter the graph data:

0 999 3 999

2 0 999 999

999 7 0 1

6 999 999 0

Shortest path matrix

0 10 3 4

2 0 5 6

7 7 0 1

6 16 9 0

**3 b. Design and implement C/C++ Program to find the transitive closure using Warshal's**

**algorithm.**

#include<stdio.h>

void warsh(int p[][10],int n)

{

int i,j,k;

for(k=1; k<=n; k++)

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

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

p[i][j]=p[i][j] || p[i][k] && p[k][j];

}

int main()

{

int a[10][10],n,i,j;

printf("\nEnter the n value:");

scanf("%d",&n);

printf("\nEnter the graph data:\n");

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

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

scanf("%d",&a[i][j]);

warsh(a,n);

printf("\nResultant path matrix\n");

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

{

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

printf("%d ",a[i][j]);

printf("\n");

}

return 0;

}

**OUTPUT**

Enter the n value:4

Enter the graph data:

0 1 0 0

0 0 0 1

0 0 0 0

1 0 1 0

Resultant path matrix

1 1 1 1

1 1 1 1

0 0 0 0

1 1 1 1