```
#include <stdio.h>
#include <limits.h>

#define V 4

void Floyd Warshall(int graph[V][V])
{
 int dist[V][V];

 for (int i = 0; i < V; i++)
 for (int j = 0; j < V; j++)
        dist[i][j] = graph[i][j];

 for (int k = 0; k < V; k++)
  for (int i = 0; i < V; i++)
    for (int j = 0; j < V; j++)
    if (dist[i][k] != INT_MAX && dist[k][j] != INT_MAX && dist[i][k] + dist[k][j] <
   dist[i][j])dist[i][j] = dist[i][k] + dist[k][j]
   ("Shortest distances between every pair of vertices:\n");
   for (int i = 0; i < V; i++) {
  for (int j = 0; j < V; j++) {
     if (dist[i][j] == INT_MAX)
        printf("INF\t");
     else
        printf("%d\t", dist[i][j]);
  }
  printf("\n");
 }
}

int main() {
 int graph[V][V] = {{0, INT_MAX, 3, INT_MAX},
            {2, 0, INT_MAX, INT_MAX},
            {INT_MAX, 7, 0, 1},{6, INT_MAX, INT_MAX, 0}};
```

```
    floydWarshall(graph);

    return 0;
}
```

Output:

Shortest distances between every pair of vertices:

```
0    10    3    4
2    0     5    6
7    7     0    1
6    16    9    0
```

Outcomes: On Completion of these Program enable students to:

- Solve Floyd's algorithm to find All-Pairs Shortest paths.
- Solve &Find Minimum Cost Spanning Tree of a given undirected graph using Floyd's **algorithm**

Viva questions

- Floyd's algorithm is an example of which type ofalgorithm.
- Difference between Floyds and Dijkstra'salgorithm.
- What is the time complexity of Floyd'salgorithm?
- What is the best,worst and average case of Floyd'salgorithm?

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

Objectives: This Program enable students to :

- Learn Warshallalgorithm to find transitive closure.
- Find path matrix graph using Warshal's **algorithm**

```
# include <stdio.h>
int n,a[10][10],p[10][10];
void path()
{
int i,j,k;
for(i=0;i<n;i++)
for(j=0;j<n;j++)
p[i][j]=a[i][j];
for(k=0;k<n;k++)
for(i=0;i<n;i++)
for(j=0;j<n;j++)
if(p[i][k]==1&&p[k][j]==1)
p[i][j]=1;
}
void main ( )
{
int i,j;
printf("Enter the number of nodes:");
scanf("%d",&n);
printf("\nEnter the adjacency matrix:\n");
for(i=0;i<n;i++)
for(j=0;j<n;j++)
scanf("%d",&a[i][j]);
 path();
printf("\nThe path matrix is shown below\n");
for(i=0;i<n;i++)
{
```

```
 for(j=0;j<n;j++)
 printf("%d ",p[i][j]);
 printf("\n");
 }
 }
```

Output:

Enter the number of nodes:4

Enter the adjacency matrix:
0 1 0 0
0 0 1 0
0 0 0 1
1 0 0 0

The path matrix is shown below
1 1 1 1
1 1 1 1
1 1 1 1
1 1 1 1

Outcomes : Aftercompletionof Programenable students to :

- Solve Warshallalgorithm to find transitive closure.
- Solve path matrix graph using Warshal's **algorithm**

Viva questions

- Warshall algorithm is an example of which type of algorithm.
- Difference between Floyds and warshall Algorithm
- What is the time complexity of warshalls algorithm?
- What is the best,worst and average case of warshall's algorithm?