Experiment 7

```
#include<stdio.h>
#define V 4
#define INF 99999
void printSolution(int dist[][V]);
void floydWarshall (int graph[][V])
{
int dist[V][V], i, j, k;
for (i = 0; i < V; i++)
for (j = 0; j < V; j++)
dist[i][j] = graph[i][j];
for (k = 0; k < V; k++)
{
for (i = 0; i < V; i++)
{
for (j = 0; j < V; j++)
{
if (dist[i][k] + dist[k][j] < dist[i][j])
dist[i][j] = dist[i][k] + dist[k][j];
}
}
}
printSolution(dist);
}
void printSolution(int dist[][V])
{
printf ("The following matrix shows the 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] == INF)
printf("%7s", "INF");
else
printf ("%7d", dist[i][j]);
}
printf("\n");
}
}
int main()
{
int graph[V][V] = { {0, 5, INF, 10},
{INF, 0, 3, INF},
{INF, INF, 0, 1},
{INF, INF, INF, 0}
};
floydWarshall(graph);
return 0;
```

```
(base) computer@computer:~/Desktop$ gcc -o AOAE7 AOAE7.c
(base) computer@computer:~/Desktop$ ./AOAE7
The following matrix shows the shortest distances between every pair of vertices
      0      5      8      9
    INF      0      3      4
    INF    INF      0      1
    INF    INF    INF      0
(base) computer@computer:~/Desktop$
```