Batch: A3	**Roll No.: 16010121045**
Experiment No: 7	
Grade: AA / AB / BB / BC / CC / CD /DD	
Signature of the Staff In-charge with date	

Title: Implementation of All Pair Shortest Path using Dynamic Programming

Objective To learn the All-Pair Shortest Path using Floyd-Warshall's algorithm

CO to be achieved:

CO 2 Describe various algorithm design strategies to solve different problems and analyse Complexity.

Books/ Journals/ Websites referred:

1. **Ellis horowitz, Sarataj Sahni, S.Rajsekaran," Fundamentals of computer algorithm", University Press**
2. **T.H.Cormen ,C.E.Leiserson,R.L.Rivest and C.Stein," Introduction to algortihtms",2nd Edition ,MIT press/McGraw Hill,2001**
3. **http://users.cecs.anu.edu.au/~Alistair.Rendell/Teaching/apac_comp3600/module4/all_pairs_shortest_paths.xhtml**
4. **https://www.geeksforgeeks.org/floyd-warshall-algorithm-dp-16/**
5. **http://www.cs.bilkent.edu.tr/~atat/502/AllPairsSP.ppt**

Theory:

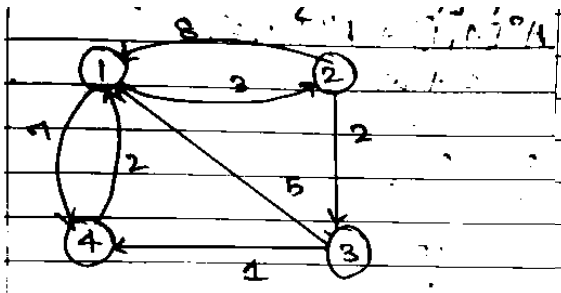
It aims to figure out the shortest path from each vertex v to every other u.

1. In all pair shortest path, when a weighted graph is represented by its weight matrix W then objective is to find the distance between every pair of nodes.
2. Apply dynamic programming to solve the all pairs shortest path.
3. In all pair shortest path algorithm, we first decomposed the given problem into sub problems.
4. In this principle of optimally is used for solving the problem.
5. It means any sub path of shortest path is a shortest path between the end nodes.

Algorithm:

```
Algorithm All_pair(W, A)
{
For i = 1 to n do
For j = 1 to n do
A [i , j] = W [i , j]
For k = 1 to n do
        {
        For i = 1 to n do
                {
                For j = 1 to n do
                {
                A [i , j] = min(A [i, j], A [i, k] + A [k, j])
                }
                }}}
```

Example:



$A^0 =$

	1	2	3	4
1	0	3	∞	7
2	8	0	2	∞
3	5	∞	0	1
4	2	∞	∞	0

$A^1 =$

	1	2	3	4
1	0	3	∞	7
2	8	0	2	15
3	5	8	0	1
4	2	5	∞	0

$A^2 =$

	1	2	3	4
1	0	3	5	7
2	8	0	2	15
3	5	8	0	1
4	2	5	7	0

$A^3 =$

	1	2	3	4
1	0	3	5	6
2	7	0	2	3
3	5	8	0	1
4	2	5	7	0

$A^4 =$

	1	2	3	4
1	0	3	5	6
2	5	0	2	3
3	3	6	0	1
4	2	5	7	0

Code:

```
#include <iostream>
#include <climits>
using namespace std;

#define V 4

void allshortestpath(int graph[][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];
                }
                if (dist[i][j] == INT_MAX)
                    cout << "INF\t";
                else
                    cout << dist[i][j] << "\t";
            }
        cout << endl;
        }
        cout<<endl;
    }
}
```

```
int main()
{
    int graph[V][V] = {{0, 15, INT_MAX, 10},
                       {INT_MAX, 0, 4, INT_MAX},
                       {INT_MAX, INT_MAX, 0, 6},
                       {INT_MAX, INT_MAX, INT_MAX, 19}};
    allshortestpath(graph);
    return 0;
}
```

```
> cd "/Users/pargatsinghdhanjal
hms/"warshal
0       15      INF     10
INF     0       4       INF
INF     INF     0       6
INF     INF     INF     19

0       15      19      10
INF     0       4       INF
INF     INF     0       6
INF     INF     INF     19

0       15      19      10
INF     0       4       10
INF     INF     0       6
INF     INF     INF     19

0       15      19      10
INF     0       4       10
INF     INF     0       6
INF     INF     INF     19
```

Analysis of algorithm:

It uses three nested loops. Innermost loop has only one statement. The complexity of that statement is Θ(1).
Running time of the algorithm is computed as

$$T(n) = \sum_{k=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \Theta(1) = \sum_{k=1}^{n} \sum_{i=1}^{n} n = \sum_{k=1}^{n} n2 = \Theta(n^3)$$

Thus, floyd's algorithm runs in cubic time.

CONCLUSION:

In this experiment, we have learnt Implementation of all Pair Shortest Path using Floyd-Warshall algorithm. We have understood the dynamic programming approach to solve all pairs shortest path problems.