Name:-Dikesh Ganboi
Roll NO:- A36

```cpp
#include <iostream>
#include <vector>

using namespace std;

#define INF 99999

// Function to perform All Pair Shortest Path using Floyd-Warshall algorithm
void floydWarshall(vector<vector<int>>& graph) {
    int V = graph.size();

    // Initialize distance matrix with graph values
    vector<vector<int>> dist = graph;

    // Update distance matrix with intermediate vertex k
    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] != INF && dist[k][j] != INF && dist[i][k] +
dist[k][j] < dist[i][j]) {
                    dist[i][j] = dist[i][k] + dist[k][j];
                }
            }
        }
    }

    // Print the shortest distances
    cout << "Shortest distances between all pairs of vertices:\n";
    for (int i = 0; i < V; i++) {
        for (int j = 0; j < V; j++) {
            if (dist[i][j] == INF) {
                cout << "INF ";
            } else {
                cout << dist[i][j] << " ";
            }
        }
        cout << endl;
    }
}

int main() {
```

```
    // Example graph represented by its adjacency matrix
    vector<vector<int>> graph = {
        {0, 3, INF, 7},
        {8, 0, 2, INF},
        {5, INF, 0, 1},
        {2, INF, INF, 0}
    };

    // Apply Floyd-Warshall algorithm
    floydWarshall(graph);

    return 0;
}
```

OUTPUT:-

```
PS C:\Users\HP\Desktop\DAA EXperiment> cd "c:\Users\HP\Desktop\DAA
EXperiment\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o
tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }

Shortest distances between all pairs of vertices:
0 3 5 6
5 0 2 3
3 6 0 1
2 5 7 0
```