**Modern Operating System and Computer Networks**

**Lab-3**

**Assignment-2**

**Date: 28-10-2025**

**Name: Chittimalla Pujitha**

**Mtech – 1st Year**

**HTNO: 2503B05118**

* **Simulation of Distance Vector Routing using the Bellman-Ford Algorithm in C++.**

#include <iostream>

using namespace std;

#define MAX 10

#define INF 100 // Large value representing no direct connection

int main() {

int cost[MAX][MAX];

int dist[MAX][MAX];

int via[MAX][MAX];

int n;

cout<< "Enter the number of nodes: ";

cin>> n;

cout<< "\nEnter the cost matrix (Enter 100 for INF):\n";

for (int i = 0; i< n; i++) {

for (int j = 0; j < n; j++) {

cin>> cost[i][j];

cost[i][j] = (cost[i][j] == 100) ?INF : cost[i][j];

dist[i][j] = cost[i][j];

via[i][j] = j;

}

}

// Bellman-Ford Update Rule

for (int k = 0; k < n; k++) { // Intermediate node

for (int i = 0; i< n; i++) { // Source node

for (int j = 0; j < n; j++) { // Destination node

if (dist[i][j] > cost[i][k] + dist[k][j]) {

dist[i][j] = cost[i][k] + dist[k][j];

via[i][j] = k; // Update via node

}

}

}

}

// Display Final Routing Tables

cout<< "\n--- Final Distance Vector Tables ---\n";

for (int i = 0; i< n; i++) {

cout<< "\nRouter " <<i + 1 << " Table:\n";

cout<< "Destination\tNext Hop\tDistance\n";

for (int j = 0; j < n; j++) {

if (i != j)

cout<< j + 1 << "\t\t" << via[i][j] + 1 << "\t\t" <<dist[i][j] <<endl;

}

}

return 0;

}

**Output:**

