**CNSL Lab Assignment 6**

**Neeti Kurulkar**

**Write a program to implement link state /Distance vector routing protocol to find suitable path  
for transmission**

**Code:**

**#include <iostream>**

**using namespace std;**

**struct node {**

**int dist[20];**

**int from[20];**

**} route[10];**

**int main() {**

**int dm[20][20], no;**

**cout << "Enter number of nodes." << endl;**

**cin >> no;**

**cout << "Enter the distance matrix:" << endl;**

**for (int i = 0; i < no; i++) {**

**for (int j = 0; j < no; j++) {**

**cin >> dm[i][j];**

**// Set distance from i to i as 0**

**dm[i][i] = 0;**

**route[i].dist[j] = dm[i][j];**

**route[i].from[j] = j;**

**}**

**}**

**int flag;**

**do {**

**flag = 0;**

**for (int i = 0; i < no; i++) {**

**for (int j = 0; j < no; j++) {**

**for (int k = 0; k < no; k++) {**

**if (route[i].dist[j] > route[i].dist[k] + route[k].dist[j]) {**

**route[i].dist[j] = route[i].dist[k] + route[k].dist[j];**

**route[i].from[j] = k;**

**flag = 1;**

**}**

**}**

**}**

**}**

**} while (flag);**

**for (int i = 0; i < no; i++) {**

**cout << "Router info for router: " << i + 1 << endl;**

**cout << "Dest\tNext Hop\tDist" << endl;**

**for (int j = 0; j < no; j++) {**

**printf("%d\t%d\t\t%d\n", j + 1, route[i].from[j] + 1, route[i].dist[j]);**

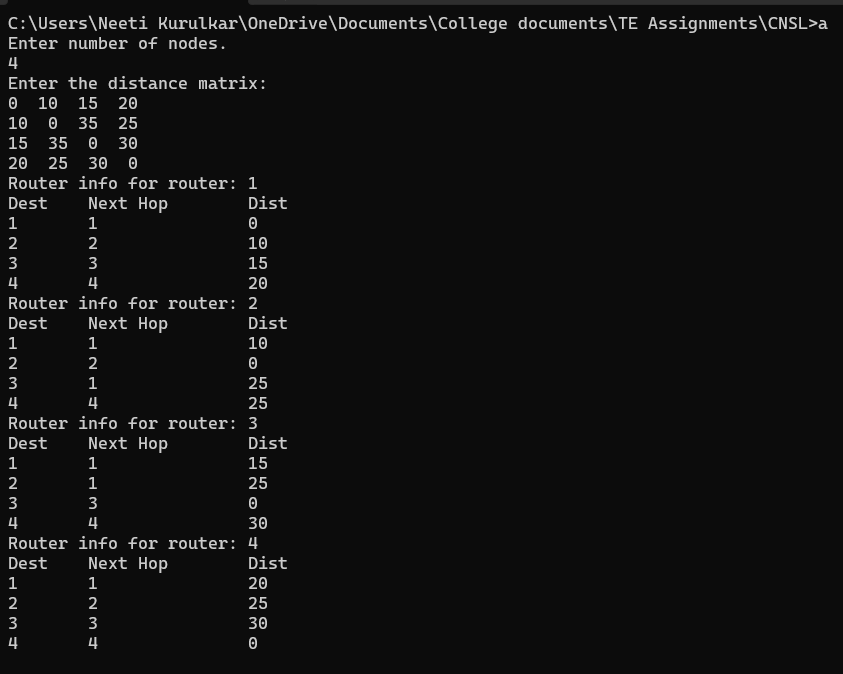
**}**

**}**

**return 0;**

**}**

**Output:**

****