```
#include<iostream>
#include<string>
using namespace std;
#define INF 99999

int adj_mat[50][50] = {0};
int visited[50] = {0};
int n;
string cities[50];

void insert() {
    cout << "Enter the number of cities: ";
    cin >> n;
    for(int i = 0; i < n; i++) {
        cout << "Enter city " << i + 1 << ": ";
        cin >> cities[i];
    }
    cout << "Enter the distance between the cities:\n";
    for(int i = 0; i < n; i++) {
        for(int j = i + 1; j < n; j++) {
            cout << "Enter the distance between " << cities[i] << " and " << cities[j] << ": ";
            cin >> adj_mat[i][j];
            if (adj_mat[i][j] == 0 && i != j) {
                adj_mat[i][j] = INF;
            }
            adj_mat[j][i] = adj_mat[i][j];
        }
    }
}

void display() {
    cout << "Adjacency Matrix (City Distances):\n";
    cout << "   ";
    for(int i = 0; i < n; i++) {
        cout << cities[i] << " ";
    }
    cout << endl;
    for(int i = 0; i < n; i++) {
        cout << cities[i] << " ";
        for(int j = 0; j < n; j++) {
            if(adj_mat[i][j] == INF) {
                cout << "INF ";
            } else {
                cout << adj_mat[i][j] << " ";
            }
        }
        cout << endl;
    }
}

void dfs(int city) {
    visited[city] = 1;
    for(int i = 0; i < n; i++) {
        if(adj_mat[city][i] != INF && adj_mat[city][i] != 0 && !visited[i]) {
            dfs(i);
        }
    }
}

bool isConnected() {
    for(int i = 0; i < n; i++) {
        visited[i] = 0;
    }
    dfs(0);
    for(int i = 0; i < n; i++) {
        if(!visited[i]) {
            return false;
        }
    }
    return true;
}

int main() {
    insert();
    display();
    if(isConnected()) {
        cout << "The flight network is connected!" << endl;
    } else {
        cout << "The flight network is not connected." << endl;
    }
    return 0;
}
```