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

const int MAX_NODES = 10;

int main() {
    int nodes, start, final;
    int distance[MAX_NODES][MAX_NODES];

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

    for (int b = 0; b < nodes; b++) {
        for (int c = b+1; c < nodes; c++) {
            cout << "Enter the distance between node " << b+1 << " and node " <<
c+1 << " : ";
            cin >> distance[b][c];
            distance[c][b] = distance[b][c];
        }
    }

    cout << "Enter the starting node : ";
    cin >> start;
    if(start<1||start>nodes)
    {
        cout<<"Starting node cannot be less than 0 or greater than number of
nodes"<<endl;
        int to_keep_exe_file_from_closing;
        cin>>to_keep_exe_file_from_closing;
        return 0;
    }
    cout << "Enter the final node : ";
    cin >> final;
    if(final>nodes||final<1)
    {
        cout<<"Final node cannot be less than 0 or greater than number of
nodes"<<endl;
        int to_keep_exe_file_from_closing;
        cin>>to_keep_exe_file_from_closing;
        return 0;
    }
    int d_distance[MAX_NODES], finaldistance;

    for (int d = 0; d < nodes; d++) {
        d_distance[d] = distance[start-1][d];
    }

    finaldistance = d_distance[final-1];

    // find the shortest distance from start to final node
    for (int i = 0; i < nodes-2; i++)
    {
        for (int j = 0; j < nodes; j++)
        {
            if (j != start-1 && j != final-1)
            {
                int new_distance = d_distance[j] + distance[j][final-1];
                if (new_distance < finaldistance) {
                    finaldistance = new_distance;
                }
            }
        }
        for (int k = 0; k < nodes; k++)
        {
            if (k != start-1 && k != final-1)
            {
                int new_distance = d_distance[k] + distance[k][final-1];
                if (new_distance < d_distance[final-1])
                {
                    d_distance[final-1] = new_distance;
                }
            }
        }
    }
    if(start==final)
    {
        cout<<"The shortest distance between "<<start<<" and "<<start<<" is :
"<<0<<endl;
    }
    else
    {
        cout << "The shortest distance between " << start << " and " << final << "
is : " << finaldistance << endl;
    }
    int to_keep_exe_file_from_closing;
    cin>>to_keep_exe_file_from_closing;
    return 0;
}
```