#include<bits/stdc++.h>

using namespace std;

#define ll long long

#define db double

#define MAX 1000000

#define rep(i,j,k) for(int i=(int)(j);i<=(int)(k);i++)

#define per(i,j,k) for(int i=(int)(j);i>=(int)(k);i--)

double Edg[1000][1000];

int path[10000][10000];

int main()

{

int k, i, j, n, m;

int t1, t2;

double x1, y1, x2, y2;

int inf = 99999999;

//初始化：

n = 32;

for (i = 1; i <= n; i++)

{

for (j = 1; j <= n; j++)

{

if (i == j) Edg[i][j] = 0;

else

Edg[i][j] = inf;

}

}

m = 46;

for (int i = 1; i <= m; i++) {

cin >> t1 >> t2 >> x1 >> y1 >> x2 >> y2;

double s = sqrt((x1 - x2)\*(x1 - x2) + (y1 - y2)\*(y1 - y2));

Edg[t1][t2] = s;

Edg[t2][t1] = s;

}

for (int i = 1; i <= n; i++) //初始化path 距离

{

for (int j = 1; j <= n; j++)

{

if (i != j && Edg[i][j] != inf) //P第0层为i

path[i][j] = i;

}

}

for (k = 1; k <= n; k++)

{

for (i = 1; i <= n; i++)

{

for (j = 1; j <= n; j++)

{

if (Edg[i][j] > Edg[i][k] + Edg[k][j])

{

Edg[i][j] = Edg[i][k] + Edg[k][j];

path[i][j] = path[k][j];

}

}

}

}

for (int i = 1; i <= n; i++) {

for (int j = 1; j <= n; j++) {

cout << Edg[i][j] << " ";

}

cout << endl;

}

while (1)

{

int x, y;

cout << "输入两个点的位置:" << endl;

cin >> x >> y;

cout << "最短距离是" << Edg[x][y] << endl;

cout << "路径是:" << endl;

cout << y;

int temp = path[x][y];

while (true)

{

if (temp == x)

{

cout << "<-" << x << endl;

break;

}

cout << "<-" << path[x][temp];

temp = path[x][temp];

}

cout << endl;

}

system("pause");

return 0;

}