

原创

【POJ 2420】A Star not a Tree? 模拟退火

2019-05-04 14:52:04

我是一只计算鸡

阅读数 47

更多

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。
本文链接：<https://blog.csdn.net/giftedpanda/article/details/89813466>

A Star not a Tree?

Time Limit: 1000MS	Memory Limit: 65536K
Total Submissions: 10577	Accepted: 4381

Description

Luke wants to upgrade his home computer network from 10mbps to 100mbps. His existing network uses 10base2 (coaxial) cables that allow connect any number of computers together in a linear arrangement. Luke is particularly proud that he solved a nasty NP-complete problem to minimize the total cable length.

Unfortunately, Luke cannot use his existing cabling. The 100mbps system uses 100baseT (twisted pair) cables. Each 100baseT cable connects devices: either two network cards or a network card and a hub. (A hub is an electronic device that interconnects several cables.) Luke has N network cards and can buy $2N-2$ network cards and connect his N computers together by inserting one or more cards into each computer and connecting them together. Or he can buy N network cards and a hub and connect each of his N computers to the hub. The first approach would require him to configure his operating system to forward network traffic. However, with the installation of Winux 2007.2, Luke discovered that network forwarding no longer worked. He couldn't figure out how to re-enable forwarding, and he had never heard of Prim or Kruskal, so he settled on the second approach: N network cards and a hub.

Luke lives in a loft and so is prepared to run the cables and place the hub anywhere. But he won't move his computers. He wants to minimize the total length of cable he must buy.

Input

The first line of input contains a positive integer $N \leq 100$, the number of computers. N lines follow; each gives the (x,y) coordinates (in mm) of a computer within the room. All coordinates are integers between 0 and 10,000.

Output

Output consists of one number, the total length of the cable segments, rounded to the nearest mm.

Sample Input

```
1 4
2 0 0
3 0 10000
4 10000 10000
5 10000 0
```

Sample Output

28284

题意：有一个由 n 个顶点的多边形，要求一个点到所有顶点的距离和最小，求最小距离。

题解：所求点为费马点，直接用模拟退火即可。

```
1 #include<cstdio>
2 #include<cmath>
3 using namespace std;
4 const double eps = 1e-10;
```



```
5 | const int maxn = 100+7; 6 | typedef struct{
7 |     double x;
8 |     double y;
9 | }point;
10 | point p[maxn];
11 | int n;
12 | double distance(point p1, point* p,int n)
13 | {
14 |     double ret = 0;
15 |     for(int i = 0; i < n; i++) ret += sqrt((p1.x-p[i].x)*(p1.x-p[i].x)+(p1.y-p[i].y)*(p1.y-p[i].y));
16 |     return ret ;
17 | }
18 | double fermentpoint(point* p, int n) // 模拟退火
19 | {
20 |     point u,v;
21 |     double x = 0, y = 0;
22 |     for(int i = 0; i < n; i++) x += p[i].x;
23 |     for(int i = 0; i < n; i++) y += p[i].y;
24 |     u.x = x/n;
25 |     u.y = y/n;
26 |     double step = 0;
27 |     for(int i = 0; i < n; i++) step += fabs(p[i].x)+fabs(p[i].y);
28 |     while(step > eps){
29 |         for(int k = 0; k < 10; k++,step/=2){
30 |             for(int i = -1; i <= 1; i++){
31 |                 for(int j = -1; j <= 1; j++){
32 |                     v.x = u.x + i*step;
33 |                     v.y = u.y + j*step;
34 |                     if(distance(u,p,n)>distance(v,p,n)) u = v;
35 |                 }
36 |             }
37 |         }
38 |     }
39 |     return distance(u,p,n);
40 | }
41 | int main()
42 | {
43 |     while(scanf("%d",&n)==1){
44 |         for(int i = 0; i < n; i++) scanf("%lf %lf",&p[i].x,&p[i].y);
45 |         double ret = fermentpoint(p,n);
46 |         printf("%.0f\n",ret);
47 |     }
48 |     return 0;
49 | }
```

有 0 个人打赏

文章最后发1024 01

©2019 CSDN 皮肤主题: 终极编程指南 设计师: CSDN官方博客

程序员节，为程序员加油！

关闭