

## P5143 攀爬者

## 题目描述

他在地形图上标记了 $N$ 个点，每个点 $P_i$ 都有一个坐标 $(x_i, y_i, z_i)$ 。所有点对中，高度值 $z$ 不会相等。HKE准备从最低的点爬到最高的点，他的攀爬满足以下条件：

- (1) 经过他标记的每一个点；
- (2) 从第二个点开始，他经过的每一个点高度 $z$ 都比上一个点高；
- (3) HKE会飞，他从一个点 $P_i$ 爬到 $P_j$ 的距离为两个点的欧几里得距离。即，

$$\sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2 + (Z_i - Z_j)^2}$$

现在，HKE希望你能求出他攀爬的总距离。

## 输入格式

第一行，一个整数 $N$ 表示地图上的点数。

接下来 $N$ 行，三个整数 $x_i, y_i, z_i$ 表示第 $i$ 个点的坐标。

## 输出格式

一个实数，表示HKE需要攀爬的总距离（保留三位小数）

## 输入样例

```
5
2 2 2
1 1 1
4 4 4
3 3 3
5 5 5
```

## 输出样例

```
6.928
```

## 解析

解题步骤如下：

- 1、按照 $z$ 轴进行高度排序
- 2、依次进行两点间的距离计算

## 编码

```
#include<bits/stdc++.h>

using namespace std;

struct Node {
    double x, y, z;

    bool operator<(const Node &other) const { //重载小于运算符
        return z < other.z;
    }
} a[50050];

int main() {
    //读入基础数据
    int n;
    scanf("%d", &n);
    for (int i = 0; i < n; i++) {
        scanf("%lf%lf%lf", &a[i].x, &a[i].y, &a[i].z);
    }
    //以重载的方式对z轴进行排序
    sort(a, a + n); //每个点高度从小到大排序
    double ans = 0;
    //累计高度
    double cx = a[0].x, cy = a[0].y, cz = a[0].z;
    for (int i = 1; i < n; i++) {
        //计算两点的欧几里得距离
        ans += sqrt(abs(cx - a[i].x) * abs(cx - a[i].x) +
                    abs(cy - a[i].y) * abs(cy - a[i].y) +
                    abs(cz - a[i].z) * abs(cz - a[i].z));

        cx = a[i].x;
        cy = a[i].y;
        cz = a[i].z;
    }
    //输出保留三位小数
    printf("%.3lf\n", ans);
    return 0;
}
```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

