

P4995 跳跳!

题目描述

你是一只小跳蛙，你特别擅长在各种地方跳来跳去。

这一天，你和朋友小 F 一起出去玩耍的时候，遇到了一堆高矮不同的石头，其中第 i 块的石头高度为 h_i ，地面的高度是 $h_0=0$ 。你估计着，从第 i 块石头跳到第 j 块石头上耗费的体力值为 $(h_i-h_j)^2$ ，从地面跳到第 i 块石头耗费的体力值是 $(h_i)^2$ 。

为了给小 F 展现你超级跳的本领，你决定跳到每个石头上各一次，并最终停在任意一块石头上，并且小跳蛙想耗费尽可能多的体力值。

当然，你只是一只小跳蛙，你只会跳，不知道怎么跳才能让本领更充分地展现。

不过你有救啦！小 F 给你递来了一个写着 AK 的电脑，你可以使用计算机程序帮你解决这个问题，万能的计算机机会告诉你怎么跳。

那就请你——会写代码的小跳蛙——写下这个程序，为你 NOIp AK 踏出坚实的一步吧！

输入格式

输入一行一个正整数 nnn ，表示石头个数。

输入第二行 nnn 个正整数，表示第 iii 块石头的高度 h_{ih_ihi} 。

输出格式

输出一行一个正整数，表示你可以耗费的体力值的最大值。

输入样例

```
2
2 1
```

输出样例

```
5
```

解析

想要获得最大的体力消耗，就需要我们在当前情况下最高和最矮的石头上来回跳。因此，排序后，不断的枚举最大最小值，即可得到答案。

注意：首次跳跃一定是从地面跳到最高的石头上！。

编码

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

using namespace std;
int n, stones[301];
//因为当前的运算含有平方，结果会非常巨大，因此需要long long
long long ans;

int main() {
    //读入数据
    cin >> n;
    for (int i = 1; i <= n; i++) {
        cin >> stones[i];
    }

    //STL升序排列
    sort(stones + 1, stones + n + 1);
    //p,q分别记录接下来要跳的石头是哪块
    //注意q要从0开始，代表从地面起跳
    int q = 0, p = n;
    //不断的在最高和最矮的石头上切换
    while (q < p) {
        //先从矮的跳到高处，计算跳一次后增加消耗的体力值
        ans += pow((stones[p] - stones[q]), 2);
        //当前站在高处，所以矮处的石头增加
        q++;
        //再从高处跳到矮处
        ans += pow((stones[p] - stones[q]), 2);
        //当前站在矮处，所以高处的石头减少
        p--;
    }
    cout << ans; //输出
    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。

