

# NOIP 模拟题

laofu

2021 年 7 月 15 日

题目名称	白云的直线	白云的商店	白云的路径	白云的花费
英文名称	line	seq	attack	new
输入文件名	line.in	seq.in	attack.in	new.in
输出文件名	line.out	seq.out	attack.out	new.out
数据组数	10	20	20	9 subtask
时间限制	1s	1s	1s	1s
空间限制	512MB	512MB	512MB	512MB

# 1 白云的直线

## 1.1 description

平面直角坐标系上有  $n$  个点，白云想用一条直线来近似他们。

白兔告诉白云，使得方差最小的直线是回归直线，若  $n$  个点为  $(x_i, y_i), i \in [1, n]$ ，则回归直线的斜率为

$$k = \frac{\sum_{i=1}^n [(x_i - x') * (y_i - y')]}{\sum_{i=1}^n (x_i - x')^2}$$

其中  $x'$  表示  $x_i$  的平均数,  $y'$  表示  $y_i$  的平均数。我们还知道, 回归直线满足  $y' = kx' + b$ , 由此也能够得到  $b$  的值。

现在白云的任务是，向平面内依次插入  $n$  个点，每插入一个点，就要求出所有点的回归直线的  $k$  和  $b$ 。

## 1.2 input

第一行一个正整数  $n$

接下来  $n$  行，每行两个实数  $x, y$  表示插入一个点  $(x, y)$

## 1.3 output

共  $n - 1$  行，每行两个实数，第  $i$  行的两个实数表示插入前  $(i + 1)$  个点的回归直线的斜率和截距。

保留四位小数。

## 1.4 样例数据

### 1.4.1 样例输入一

3

7.1 9.99

10.2 30.18

6.2 1.89

### 1.4.2 样例输出一

6.5129 -36.2516

6.9280 -40.2497

## 1.5 数据范围

对于 60% 的数据,  $n \leq 10^3$

对于 100% 的数据,  $n \leq 10^5, x, y \leq 100$

给出的实数的小数位数不超过 6 位

## 2 白云的商店

### 2.1 description

白云建立了  $n$  个商店，白兔打算按照编号  $1 \sim n$  的顺序访问这些商店。商店  $i$  有一个价格  $a_i$  表示交易商品所需的代价。

白兔在按顺序走时，每到达一个商店，可以花费代价购买一件商品，并放入自己手中。也可以出售手上的商品，并获得利润。

白兔的力量有限，同一时刻只能携带一个商品。问它遍历完所有商店后能够获得的利润最大是多少？

白兔的精力也有限，所以，在最大化利润的前提下，它想让交易次数尽可能地少。

当然，白云不想让白兔轻松获利，它有时会命令一段区间内的商店把价格同时加上一个数。

### 2.2 input

第一行一个整数 表示数据组数

对于每一组数据

第一行两个整数  $n, Q$

第二行  $n$  个整数描述序列  $a_1 \sim a_n$

接下来  $Q$  行，每行第一个数表示操作类型。

如果为 1 表示你需要帮助白兔求出，如果它立即出发，它能获得的最大利润，以及在最大利润的前提下，最小交易次数。

如果为 0 后面接三个整数  $l, r, c$  表示在区间  $[l, r]$  内的商品价格都加上  $c$ 。

### 2.3 output

对于每一个询问操作，输出一行两个数表示最大利润和最小交易次数。

## 2.4 测试样例

### 2.4.1 样例输入一

```
2
5 9
9 10 7 6 8
1
0 4 5 2
0 3 5 4
1
0 2 5 -2
0 3 5 -3
0 4 5 -2
0 5 5 -4
1
4 3
2 4 3 5
1
0 3 3 3
1
```

### 2.4.2 样例输出一

```
3 4
5 2
0 0
4 4
4 2
```

### 2.4.3 样例解释

对于第一个询问 5 个数为 9 10 8 6 8

我们会选择在 1,2,4,5 进行交易，收益为 $-9+10-6+8=3$

对于第二个询问 5 个数为 9 10 11 12 14

我们会选择在 1,5 两个店交易，收益为 $-9+14=5$

对于第三个询问 5 个数为 9 8 6 5 3

交易不会带来任何利润，所以什么都不用做

## 2.5 数据范围

对于所有数据满足  $T \leq 5, n, Q, \leq 10^5, |c| \leq 10^5$ ，任意时刻  $0 < a_i < 2^{31}$

测试点编号	$n \leq$	$Q \leq$	初始 $a_i$	其它
1,2	13	13	1000	
3,4,5,6	1000			
7,8,9,10	40000	$10^5$		$T = 1$
11 ~ 20	$10^5$	$10^5$		

## 2.6 评分方式

对于每一个测试点，回答第一问正确可以获得 4 分，回答第二问正确可以获得 1 分。

但是无论如何，对于每一个询问你的程序都必须恰好输出两个数，否则直接记 0 分

### 3 白云的路径

#### 3.1 description

白云有一颗  $n$  个节点的树，每个点有一个点权

白兔需要找到两条点不相交的路径，要求最大化这两条路径覆盖点的点权和

#### 3.2 input

第一行包含一个数  $n$  表示树的结点数

接下来一行  $n$  个数表示每个点的点权

接下来  $n - 1$  行每行两个数描述一条树边。

#### 3.3 output

输出最大点权和

#### 3.4 样例数据

##### 3.4.1 样例输入一

```
8
1 2 99 4 5 3 2 1
1 2
1 4
1 6
1 3
4 5
6 7
6 8
```

##### 3.4.2 样例输出一

```
115
```

### 3.5 数据范围

测试点编号	$n \leq$	特点	测试点编号	$n \leq$	特点
1, 2, 3	10	保证数据随机	11...15	$10^5$	每个点的度数 $\leq 6$
4, 5, 6	1000		16...20		
7, 8, 9, 10	$10^5$				



## 4 白云的花费

### 4.1 description

白云在白兔的城市，打算在这里住  $n$  天，这  $n$  天，白云计划找白兔订购水。

白云给出了接下来的  $n$  天，每天的需水量，第  $i$  天为  $D_i$  升。

白兔给出了接下来的  $n$  天，每天水的价格，第  $i$  天为  $P_i$  元每升。

白云的消去有一个共用水壶，最大容量为  $V$  升，初始为空。

接下来每天，白云可以进行如下操作：

1. 把水壶中的水使用掉
2. 向白兔购买若干水，并放入水壶中
3. 向白兔购买若干水，并使用

任何时候水壶中的水不能超过  $V$  升，而且每升水每在水壶中存放一天，需要付出  $m$  元

白兔为了难为白云，决定在某些天进行破坏操作，即选择一个子序列  $b_1 \cdots b_t$ ，在这序列中的每一天，它会在当天早上把前一天储存的水放掉。第  $i$  天有一个破坏难度  $val_i$ ，白兔为了挑战自己，决定让自己进行破坏操作的  $val$  是严格单调递增。它会找一个破坏天数最多的方案，在保证破坏次数最多的前提下，使得破坏序列的字典序最小。

白云已经知道了白兔的想法，并且获取了这个破坏难度，所以它已经能计算出白兔会在哪些日子进行破坏。

那么白云想知道，在此基础上，白云要度过接下来  $n$  天的最小总花费。

### 4.2 input

第一行三个整数  $n, m, V$

第二行  $n$  个整数描述  $val$  数组

第三行  $n$  个整数描述  $D$  数组

第四行  $n$  个整数描述  $P$  数组

### 4.3 output

输出最小花费

### 4.4 测试样例

#### 4.4.1 样例输入一

```
5 10 10
5 1 2 0 3
2 2 2 2 2
1 10 20 50 10
```

#### 4.4.2 样例输出一

```
142
```

#### 4.4.3 样例解释

破坏序列长度为 3，其中字典序最小的为  $\{2, 3, 5\}$

2,3,5 这三天需要的水必须当天购买

最优方案为  $2 * 1 + 2 * 10 + 2 * 20 + 2 * 30 + 2 * 10 = 142$

只有第 4 天的水是水壶中存的水

## 4.5 数据范围

subtask	分值	$n \leq$	$d_i, V \leq$	$val \leq$
1	30	2	$10^7$	$10^4$
2	20	$10^6$	20	
3	20	200	$10^7$	
4	15	1000	1000	$10^9$
5	3	$10^4$	$10^4$	
6	3	$5 * 10^4$		
7	3	$10^5$		
8	3	$10^6$	$10^5$	
9	3		$10^7$	

所有数据满足  $n \leq 10^6, m, p_i \leq 1000$

答案  $\leq 2^{63} - 1$

## 4.6 温馨提示

此题输入量较大，建议采用读入优化