

方差 (math.cpp T1)

时间限制 1.00s 空间限制 128MB

题目背景

小文同学喜欢吃糖，小文希望买一些糖回来。

题目描述

总共有 n 颗糖，第 i 颗糖的价格为 a_i 。小文希望购买 m 颗糖，而小文需要支付的总金额为这 m 颗糖的方差。现在问小文购买这 m 颗最少需要支付多少钱。

注意，在本题中，方差的定义为：对于 k 个数 b_1, b_2, \dots, b_k ，设其平均数为 b ，则方差为 $((b_1 - b)^2 + (b_2 - b)^2 + \dots + (b_k - b)^2) \div k$ 。

输入格式

第一行两个整数 n, m 。

接下来一行 n 个整数代表 a_1, a_2, \dots, a_N 。

输出格式

一行一个四位小数代表答案。

样例 #1

样例输入 #1

```
5 3
1 100 2 100 3
```

样例输出 #1

```
0.6667
```

样例 #2

样例输入 #2

```
20 16
79013 45517 40463 15281 69586 88636 2540 52103 76573 2893 60890 80639 70045
39320 9023 27667 63305 82556 12775 63716
```

样例输出 #2

提示

测试点	$n \leq$	$m \leq$	$a_i \leq$
1 ~ 4	20	n	10^5
5 ~ 6	10^5	2	10^5
7 ~ 8	10^5	20	10^5
9 ~ 10	10^5	n	10^5

取糖 (candy.cpp)

时间限制 1.00s 空间限制 128MB

题目背景

众所周知，小文同学喜欢吃糖。

题目描述

小文将买来的糖放进了冰箱冷藏，但是小文想吃糖了，小文希望把自己想吃的糖从冰箱里面拿出来。

具体来说，小文同学的冰箱是一棵 n 个点的树，每个点有一颗糖，第 i 个点的糖的美味值是 a_i 。小文每次取糖会从根节点出发，指定一个目标节点 p ，走到 p 点并且把这条路径上的所有糖取走。但小文不满足只走到 p ，所以接下来小文会继续从 p 出发去取其他的糖。但是由于小文的冰箱的特殊构造，一条边一旦被走过一次就不能再走了，所以小文要仔细计划如何行动。因此，小文会有 m 次询问，每次询问给定根节点 q 和目标节点 p ，小文想知道在这种情况下她能取走的糖果的美味值之和是多少。

输入格式

第一行两个整数 n, m 。

第二行 n 个整数代表 a_i 。

接下来 $n - 1$ 行每行两个整数代表树上的一条边。

接下来 m 行每行两个整数 q, p 代表一次询问。

输出格式

输出 m 行每行一个整数代表答案。

样例 #1

样例输入 #1

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

样例输出 #1

```
11
18
```

提示

测试点	$n \leq$	$a_i \leq$	$m \leq$	$p \leq$	$q \leq$
1 ~ 2	10^3	10^5	10^3	n	n
3 ~ 4	10^5	10^5	10^5	1	n
5 ~ 6	10^5	10^5	10^5	n	1
7 ~ 10	10^5	10^5	10^5	n	n

内卷 (lift.cpp)

时间限制 1.00s 空间限制 512MB

题目背景

小文是个爱学习的同学，放暑假了，想到同学家里去问一道题。

同学住在一栋没有地下室的很高的楼里面，于是上楼就成了一个麻烦事。

但更麻烦的事来了，小文并不知道同学住在那一层，只知道她住在这栋楼里，于是她只能一层一层找。

题目描述

同学家的楼有 h 层，有一台电梯，但这个电梯十分奇怪。

一共有 $n + 1$ 个按键：

对于按键 i ：满足当前楼层 $\leq h - a_i$ ，按下后会向上走 a_i 层。

还有一个按键，按下可以回到第一层。

也就是说任意时刻小文的楼层数必须 $\leq h$ 且 ≥ 1 。

此时小文正在第一层，她可以按电梯按键无数次，求她最多能到达的楼层的数量（注意，**不存在第 0 层**）。

一共有两种询问：

$m = 1$ 时，只输出能到达的楼层数。

$m = 2$ 时，第一行输出能到达的楼层数，第二行**从小到大**输出所有能到达的楼层。

输入格式

一行三个正整数 h, m, n 。

第二行 n 个整数， $a_1, a_2 \dots a_n$ ，表示 n 个按键。

输出格式

$m = 1$ 时，只输出能到达的楼层数（含第一层）。

$m = 2$ 时，第一行输出能到达的楼层数（含第一层），第二行从小到大输出所有能到达的楼层。

样例 #1

样例输入 #1

```
15 2 3
4 7 9
```

样例输出 #1

```
9
1 5 8 9 10 12 13 14 15
```

样例 #2

样例输入 #2

```
3333333333 1 3
99005 99002 100000
```

样例输出 #2

```
33302114671
```

提示

测试点	$h \leq$	$m =$	$n \leq$	$a_i \leq$
1 ~ 3	10	2	10	10
4 ~ 7	10^5	2	3	10^4
8 ~ 9	10^9	1	100	2
10 ~ 15	10^{12}	1	100	10^5
16 ~ 20	$2^{63} - 1$	1	100	10^9

数据保证 $1 \leq a_1 \leq 10^5$, $0 \leq a_i \leq 10^9 (i \geq 2)$ 。

软糖 (flavour.cpp)

时间限制 1.00s 空间限制 512MB

题目背景

小文同学很爱吃糖，特别是软糖。

题目描述

小文同学有一排软糖，她可以任意选一段区间吃掉。

如果她吃到的相邻的两个软糖的味道不一样，她的愉悦值就会加 k 。同时她还想知道她一共吃了多少个口味为 y 的软糖。

但有时小文会交换相邻的两个软糖。

小文很喜欢吃软糖，所以她一定会思索再三再吃（没有真的吃，只是询问），因此会有 q 个操作。

输入格式

第一行一个正整数 n ，表示一共有 n 个软糖。

第二行 n 个整数， $a_1, a_2 \dots a_n$ ，表示每个软糖的味道。

接下来一个正整数 q ，表示有 q 次询问。

每个询问一行，第一个数为 op 。

若 $op = 1$ ，后面还会有四个整数 l, r, k, y ，表示如果她吃掉 l 到 r 区间的所有糖果，且相邻两个糖果味道不同时愉悦值会加 k ，问这段区间味道为 y 的软糖的个数。

若 $op = 2$ ，后面会有一个正整数 x ，表示第 x 个糖果与第 $x + 1$ 个糖果交换了位置。

输出格式

对于每一次询问，输出一行两个整数，用空格分开，分别表示愉悦值和一共吃了多少个口味为 y 的软糖。

样例 #1

样例输入 #1

```
9
1 1 4 5 1 4 1 1 9
5
1 1 4 1 1
2 2
1 1 4 1 4
2 5
1 5 9 2 1
```

样例输出 #1

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

提示

样例解释：

[1, 1, 4, 5]中有两个味道为 1 的糖果，有两处相邻的糖果味道不同， $2 \times k = 2 \times 1 = 2$ 因此输出 2 和 2。

交换 2 和 3 位置上的糖果，变为 [1, 4, 1, 5, 1, 4, 1, 1, 9]。

[1, 4, 1, 5]中有一个味道为 4 的糖果，有三处相邻的糖果味道不同， $3 \times k = 3 \times 1 = 3$ 因此输出 3 和 1。

交换 5 和 6 位置上的糖果，变为 [1, 4, 1, 5, 4, 1, 1, 1, 9]。

[4, 1, 1, 1, 9]中有三个味道为 1 的糖果，有两处相邻的糖果味道不同， $2 \times k = 2 \times 2 = 4$ 因此输出 4 和 3。

测试点	$n \leq$	$q \leq$	$a_i \leq$	$k \leq$
1 ~ 3	10	20	5	1
4 ~ 5	10^3	10^5	10^5	1
6 ~ 7	10^5	1	10^5	10^3
8 ~ 10	10^5	10^5	5	10^3
11 ~ 20	4×10^5	10^5	10^5	10^5

数据保证 $1 \leq x \leq n - 1$, $1 \leq a_i \leq n$, $1 \leq l \leq r \leq n$ 。