

拼点游戏

【问题描述】

小 W 和小 Y 都很喜欢玩一种“拼点游戏”。游戏中两个人分别通过某种操作产生一个数作为自己的“点数”，点数大的一方获胜。“拼点游戏”的规则如下：

- 1、游戏开始时，给定一个包含 n 个元素的正整数序列 $U = (u_1, u_2, \dots, u_n)$ 。
- 2、定义 U 的一个下标序列 $I = (i_1, i_2, \dots, i_m)$ 是满足 $1 \leq i_1 < i_2 < \dots < i_m \leq n$ 的一个整数序列 (m 可以为 0，即序列可以为空)，并且其对应 U 的子序列为 $V = (u_{i_1}, u_{i_2}, \dots, u_{i_m})$ 。

- 3、定义下标序列 $I = (i_1, i_2, \dots, i_m)$ 对应的点数 $D(I)$ 为

$$D(I) = \sum_{p=1}^m u_{i_p} * (-1)^p$$

- 4、进行游戏时两人分别选择一个下标序列，谁选择的下标序列对应的点数 $D(I)$ 大，谁就获胜。

然而在每次游戏中，小 W 总是能准确无误的算出点数最大的最优下标序列。为了让游戏更加具有竞技性，他们制定了下列额外规则：

Ex1. 小 W 可以选择一个非空区间 $[l, r]$ ，并将 u_l, u_{l+1}, \dots, u_r 同时增加一个整数 c ，产生的新序列将取代原序列 U 。

Ex2. 当他们对于当前的 U 序列进行一次“拼点游戏”时，允许小 Y 在小 W 选出最优下标序列 $I_W = (i_1, i_2, \dots, i_m)$ 之后，对 I_W 进行任意次修改操作。每次修改操作规则如下：

- (1) 任意选择一个正整数 k 满足 $2k + 1 \leq m$ ，以及两个非负整数 z_1, z_2 满足 $i_{2k} + z_1 < i_{2k+1} - z_2$ ；
- (2) 将 i_{2k} 修改为 $i_{2k} + z_1$ ，将 i_{2k+1} 修改为 $i_{2k+1} - z_2$ 。

若小 W 选出的下标序列 I_W 经过小 Y 若干次修改操作之后所对应的点数小于等于 0，则小 Y 获胜。

现在给出小 W 所进行的 Ex1 操作的信息，请你对于每一次“拼点游戏”，帮助他们计算：

- a) 小 W 一开始所能选出的最优下标序列对应的点数是多少？
- b) 小 Y 最少需要进行几次修改操作才能获胜？即使得 $D(I_W) \leq 0$ 。

【输入格式】

输入文件 joy.in 的第一行包含一个正整数 T ，表示测试数据的组数。接下来为 T 组数据。

每一组数据的第一行包含两个整数 n 和 q ，分别表示 U 中的元素个数和事件个数。

接下来的一行，包含 n 个用一个空格隔开的正整数，第 i 个整数为初始的序列中第 i 个元素 u_i 。

接下来 q 行，每行代表一个事件（按事件发生顺序输入）。每行的第一个数非 0 即 1，表示这个事件的类型。

若为 0：在 0 之后还有三个整数 l ， r 和 c （这四个数之间均有一个空格），表示小 W 将 u_l, u_{l+1}, \dots, u_r 增加 c ；

若为 1：表示两人进行了一次“拼点游戏”，你需要输出相应的结果。

输入数据保证序列 U 中的所有元素总是正整数。

【输出格式】

输出文件为 joy.out。

对于每一组测试数据，依次对每一次“拼点游戏”输出一行包含两个由一个空格隔开的整数 D_{\max} 和 X ，其中

D_{\max} 为对于 当前序列 U ，小 W 所能选出的最优下标序列所对应的点数；

X 表示小 Y 最少需要进行几次修改操作才能获胜。如果小 Y 不论多少次操作都无法获胜，则输出 -1。

数据保证最优下标序列总是唯一的。

【评分标准】

一个测试点包含多组测试数据，对于该测试点：

如果所有的 D_{\max} 均正确但某个 X 不正确，则可以得到 3 分；

如果所有的 X 均正确但某个 D_{\max} 不正确，则可以得到 7 分；

如果所有回答均正确，则可以得到 10 分。

【样例输入】

```

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

```

【样例输出】

```
3 1
5 -1
0 0
4 -1
4 -1
```

【样例说明】

输入数据包含两组测试数据。

在第一组测试数据中：

第一次“拼点游戏”时，最优下标序列为(1,2,4,5)，小 Y 只需要进行一次修改操作：选择 $k=1$ ，以及非负整数 $z_1=1$ ， $z_2=0$ 。这样经过修改操作之后下标序列将变为(1,3,4,5)，小 Y 获胜。

第三次“拼点游戏”时，序列 U 为(9,8,6,5,3)，小 W 所选择的最优下标序列为空序列，所产生的点数为 0。在这种情况下，小 Y 无法进行任何修改操作（也无需进行任何修改操作），此时小 Y 已经直接获胜。

【数据规模】

对于 10% 的数据满足 $n, q \leq 13$ ；

对于 30% 的数据满足 $n, q \leq 1000$ ；

对于另外 20% 的数据满足 $T=1$ 且 $n \leq 40000$ ；

对于 100% 的数据满足 $T \leq 3$ 且 $n, q \leq 10^5$ ，同时初始序列 U 满足 $0 < u_i < 2^{31}$ ， $|c| < 10^5$ 。