

Problem 4 - Candies (Programming) (10 points)

Problem Description

Baluteshih brings N candies to his friend, Waynetu. Those candies are lined on the table. Each candy has its own sweetness indicated by an integer. The sweetness of the candies from left to right are a_1, a_2, \dots, a_N , respectively.

Baluteshih assigns Waynetu an interesting mission. He asks Waynetu to remove some candies from the table, so that the remaining candies on the table are *alternating*. Formally, we say that a sequence of candies with sweetness b_1, b_2, \dots, b_k is *alternating* if $b_i \times b_{i+1} \leq 0$ holds for all $1 \leq i < k$.

With the aforementioned rule, Waynetu hopes to maximize the sum of the sweetness of the candies on the table. Please help Waynetu find the maximum possible sum of the remaining candies' sweetness and provide a solution to reach the optimal case.

Input

The first line contains two integers T , representing the number of testcases, and $flag$ ($flag \in \{0, 1\}$), which will be described in the output section.

Each testcase includes two lines: the first line contains an integer N ($1 \leq N \leq 10^5$), and the second line contains N integers a_1, a_2, \dots, a_N ($|a_i| \leq 10^9$).

It is guaranteed that the sum of N does not exceed 10^5 .

Test Group 0 (0 %)

- Sample Input.

Test Group 3 (10 %)

- $flag = 1$.
- $\sum N \leq 1000$.

Test Group 1 (20 %)

- $flag = 0$.
- $\sum N \leq 1000$.

Test Group 4 (20 %)

- $flag = 1$.

Test Group 2 (50 %)

- $flag = 0$.

Output

For each testcase, please print an integer representing the maximum possible sum of the sweetness in the first line. If $flag$ equals 1 mentioned in the input section, please furthermore print out one optimal way in the second line: an integer k representing the number of candies remaining on the table, followed by k integers i_1, i_2, \dots, i_k , representing the indices of candies. *If there are multiple optimal solutions, you may print out any one of them.*

Sample Input 1

```
1 0
5
3 -1 6 -7 4
```

Sample Output 1

```
8
```

Sample Input 2

```
2 0
3
1 2 3
4
1 -2 3 -4
```

Sample Output 2

```
3
3
```

Sample Input 3

```
3 1
1
-1
3
5 0 1
4
-1 -2 -3 -4
```

Sample Output 3

```
-1
1 1
6
3 1 2 3
-1
1 1
```

Hint

1. Since each input includes several independent testcases, please carefully clear all results of the current testcase before dealing with the next testcase.