Alawn's Problem

Time limit: 3.0s **Memory limit:** 256M

Alawn has a length N array of pairs (a_i, b_i). Initially, $a_i = b_i$ for all i.

Alawn loves arrays that satisfy the following property:

- The a values can be sorted by only swapping elements i and j where $b_i \times b_j \leq V$.

A *modification* to the array consists of decreasing element i's b_i value by 1. Can you determine the minimum number of modifications required to turn the array of pairs into one that Alawn loves?

Input Specification

The first line will contain T $(1 \leq T \leq 10)$, the number of test cases. T test cases follow.

For each test case, the first line will contain two integers N, V $(1 \leq N \leq 10^5, 1 \leq V \leq 10^{18})$, the number of elements in the array and the special value V.

The second line will contain N integers, a_i $(1 \leq a_i \leq 10^9)$, the elements of the array. **Recall that $a_i = b_i$ initially.**

Output Specification

For each case, output the minimum number of modifications required on its own line.

Sample Input

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

Sample Output

```
0
1
```