Competitive Programming SS24

Submit until end of contest

Problem: recommend (1.0 second timelimit)

In your previous project, you build a tool that could estimate the difficulty of competitive programming problems. You don't quite know how it works, but it assigns problems a rating between 1 and 10^6.

Next, you want to build a problem recommender based on these ratings. A user should be able to input a difficulty u and get a recommended problem to solve next. You plan is to first select a subset of suitable problems, and to then choose one of them uniformly at random. You have two criteria for this subset:

- To suggest varied problems, the subset should be as large as possible.
- In expectation, the suggested problem should have a difficulty of at least u, so that the user doesn't get bored.

Given a set of problems and a difficulty u, what is the size of the largest subset you can choose?

Input The first line contains n ($1 \leq n \leq 2 \cdot 10^5$), the number of problems, and u ($1 \leq u \leq 10^6$), the difficulty input by the user. The next line contains n numbers d_i ($1 \leq d_i \leq 10^6$), the difficulties of the problems in your collection.

Output Output the size of the largest subset of problems you can choose, such that, in expectation, a uniformly chosen problem from the subset has a difficulty of at least u.

Sample input

```
2 2
2 1
```

```
5 10
14 3 10 4 17
```

Sample output

```
1
```

```
4
```