Constructing Pyramid

locked

Problem Submissions Leaderboard Discussions

You are given n cuboidal boxes $b_1, b_2, \ldots, b_n$ with square bases, having dimensions $(a_1, a_1, 1), (a_2, a_2, 1), \ldots, (a_n, a_n, 1)$ respectively, where a_i is the length of each side of the square base for the box b_i $(1 \leq i \leq n)$. Note that the height of each of the n boxes is 1. So, a pyramid of height h can be constructed by stacking h such boxes on top of each other, in strictly decreasing order of size; that is, for any two boxes b_p and b_q (representing the boxes at height p and q respectively) in a pyramid where $i > j$ should have side lengths $a_i < a_j$.

Given n, h and a_i, print the maximum number of pyramids of height h that can be constructed.

Notes:

- You have to pick boxes without replacement.
- Only the most efficient solutions will pass all test cases, while others will pass only a subset of test cases.
- In C++, use fast I/O by adding `ios_base::sync_with_stdio(false); cin.tie(NULL);` as the first line within the main() function.

Input Format

First line contains one integer t indicating number of test cases. After which, for each test case,

- first line contains two integers n and h.
- second line contains n sorted space seperated integers a_i representing the side lengths of the n boxes.

Constraints

$1 \leq t \leq 10$

$1 \leq n \leq 10^6$

$1 \leq h \leq 10^9$

$1 \leq a_i \leq 10^9$

Output Format

For each test case, print a single line containing a single integer - the maximum number of pyramids possible.

Sample Input 0

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

Sample Output 0

```
2
0
```

Explanation 0

In test case 1, maximum of 2 pyramids of height 2 can be constructed {1,2} and {2,4}. In test case 2, no pyramid of height 3 can be constructed.

Submissions: 274
Max Score: 100
Difficulty: Medium

Rate This Challenge:

More

Current Buffer (saved locally, editable) C++

```
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code