

# Angry Professor



Problem Submissions Leaderboard Discussions Editorial 🗎 Topics

A Discrete Mathematics professor has a class of students. Frustrated with their lack of discipline, he decides to cancel class if fewer than some number of students are present when class starts. Arrival times go from on time ( $arrivalTime \le 0$ ) to arrived late (arrivalTime > 0).

Given the arrival time of each student, determine if the class is canceled.

## **Input Format**

The first line of input contains t, the number of test cases.

Each test case consists of two lines.

The first line has two space-separated integers, n (students in the class) and k (the cancelation threshold). The second line contains n space-separated integers  $(a_1, a_2, \ldots, a_n)$  describing the arrival times for each student.

**Note:** Non-positive arrival times ( $a_i \leq 0$ ) indicate the student arrived early or on time; positive arrival times ( $a_i > 0$ ) indicate the student arrived  $a_i$  minutes late.

# **Constraints**

- $1 \le t \le 10$
- $1 \le n \le 1000$
- $1 \leq k \leq n$
- $-100 \le a_i \le 100, where \ i \in [1, n]$

### **Output Format**

For each test case, print the word YES if the class is canceled or NO if it is not.

#### Note

If a student arrives exactly on time  $(a_i = 0)$ , the student is considered to have entered before the class started.

#### Sample Input

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

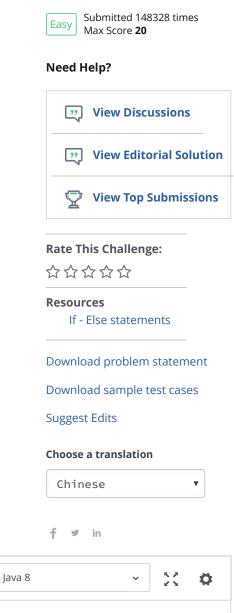
# **Sample Output**

YES NO

# **Explanation**

For the first test case, k=3. The professor wants at least 3 students in attendance, but only 2 have arrived on time (-3 and -1) so the class is canceled.

For the second test case, k=2. The professor wants at least 2 students in attendance, and there are 2 who have arrived on time (0 and -1) so the class is *not* canceled.



1 v import java.io.\*;
2 import java.util.\*;
3 import java.text.\*;

**Current Buffer** (saved locally, editable) ♀ • •

```
import java.math.*;
   import java.util.regex.*;
7 ▼ public class Solution {
8
        static String angryProfessor(int k, int[] a) {
9 ▼
10 ▼
            for (int time:a) {
                if (time <= 0) { k--; }
11 ▼
12
13
            return (k > 0) ? "YES" : "NO";
14
        }
15
16 ▼
        public static void main(String[] args) {
17
            Scanner in = new Scanner(System.in);
18
            int t = in.nextInt();
            for(int a0 = 0; a0 < t; a0++){
19 ▼
                int n = in.nextInt();
20
21
                int k = in.nextInt();
22 ▼
                int[] a = new int[n];
                for(int a_i = 0; a_i < n; a_i++){
23 ▼
                    a[a_i] = in.nextInt();
24 ▼
25
                }
26
                String result = angryProfessor(k, a);
                System.out.println(result);
27
28
            in.close();
29
30
        }
31
   }
32
                                                                                                 Line: 32 Col: 1
```

<u>**1**</u> <u>Upload Code as File</u> ☐ Test against custom input

Run Code

Submit Code

# Congrats, you solved this challenge!

Challenge your friends: f 💆 in

✓ Test Case #0

Test Case #1

✓ Test Case #2

✓ Test Case #3

✓ Test Case #4

✓ Test Case #5

✓ Test Case #6

✓ Test Case #7

✓ Test Case #8

Test Case #9

✓ Test Case #10

✓ Test Case #11

Next Challenge

You've earned 20.00 points. You are now 83.88 points away from the 4th star for your Problem Solving badge.

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature