# Angry Professor

by devuy11



A Discrete Mathematics professor has a class of $N$ students. Frustrated with their lack of discipline, he decides to cancel class if fewer than $K$ students are present when class starts.

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**

- $1 \leq T \leq 10$
- $1 \leq N \leq 1000$
- $1 \leq K \leq N$
- $-100 \leq a_i \leq 100, where\ i \in [1, N]$

**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 $-1$). Thus, 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 $-1$). Thus, the class is *not* canceled.

Submissions:141466

Max Score:20
Difficulty: Easy

Rate This Challenge:

Need Help?

If - Else statements

More

Current Buffer (saved locally, editable) Java 7

```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        int t = in.nextInt();
        for(int a0 = 0; a0 < t; a0++){
            int n = in.nextInt();
            int k = in.nextInt();
            int a[] = new int[n];
            for(int a_i=0; a_i < n; a_i++){
                a[a_i] = in.nextInt();
            }
        }
    }
}

```

Line: 1 Col: 1

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