



Islamic University of Technology
Department of Computer Science and Engineering

Lab 2: Sorting

CSE 4404: Algorithms Lab
Summer 2023-24

Task A. The Great Container Fill-Off

Time Limit: 1 second | Memory Limit: 256 MB

You are given an empty container that can hold exactly x liters of water. You are also given n bottles, each containing a certain amount of water. From each bottle, you may pour **any amount** of water into the container until it is empty.

Your goal is to fill the container with **exactly** x liters of water using water from the bottles. You want to use the **minimum number of bottles** from which you pour water (regardless of how much is poured from each).

If it is not possible to collect exactly x liters, output -1 .

Input Format

The first line contains a single integer t ($1 \leq t \leq 10^5$) — the number of test cases.

The first line of each test case contains two integers n and x ($1 \leq n \leq 10^4$, $1 \leq x \leq 10^9$) — the number of bottles and the exact capacity of the container in liters.

The second line of each test case contains n space-separated integers a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) — the amount of water available in each bottle.

The sum of n over all test cases does not exceed 10^5 .

Output Format

For each test case, print a single integer in a line — the minimum number of bottles required to collect exactly x liters of water, or -1 if it is impossible.

Examples

Sample Input	Sample Output
4	3
5 10	2
2 3 4 5 1	-1
4 7	3
5 3 4 2	
3 5	
1 1 1	
4 12	
4 6 3 1	

Task B. Say Cheese!

Time Limit: 1 second | Memory Limit: 256 MB

Mark is asked to take a group photo of $2n$ people. The i -th person has height h_i units. To take the photo, he wants to divide the people into two rows: a **front row** and a **back row**, each consisting of exactly n people.

To ensure that everyone is visible, Mark wants the j -th person in the back row from the left to be at least x units taller than the j -th person from the left in the front row, for each $1 \leq j \leq n$.

Help Mark determine whether it is possible to assign the people into two such rows so that the above condition is satisfied.

Input Format

The first line contains two integers n and x ($1 \leq n \leq 10^5$, $1 \leq x \leq 10^9$) — the number of people in each row, and the minimum required height difference between corresponding people in the back and front rows.

The second line contains $2n$ integers h_1, h_2, \dots, h_{2n} ($1 \leq h_i \leq 10^9$) — the heights of all people.

Output Format

Print a single line containing **YES** if it is possible to arrange the people into two rows satisfying the condition, or **NO** otherwise.

Examples

Sample Input	Sample Output
3 6 1 3 9 10 12 16	YES
3 1 2 5 2 2 2 5	NO
1 2 8 6	YES

Task C. Plan It Right

Time Limit: 1 second | Memory Limit: 256 MB

You are an event planner hired to organize a series of activities for a group of tourists visiting a city. You are given a list of n activities, each with a start time and an end time. A tourist can participate in an activity only if they are not already engaged in another one at that time.

Your goal is to select a subset of non-overlapping activities such that the tourists can attend the **maximum number of activities**.

Input Format

The first line contains a single integer t ($1 \leq t \leq 10^4$) — the number of test cases.

For each test case, the first line contains a single integer n ($1 \leq n \leq 10^5$) — the number of activities.

The second line contains n space-separated integers s_1, s_2, \dots, s_n ($1 \leq s_i \leq 10^9$) — the start times of the activities.

The third line contains n space-separated integers e_1, e_2, \dots, e_n ($1 \leq e_i \leq 10^9, s_i \leq e_i$) — the end times of the activities.

The sum of n over all test cases does not exceed 10^5 .

Output Format

For each test case, print a single integer — the maximum number of non-overlapping activities that the tourists can attend.

Examples

Sample Input	Sample Output
5	3
4	2
0 5 1 3	2
6 7 2 4	2
3	2
1 4 6	
5 7 10	
6	
3 8 2 1 5 3	
9 9 5 4 9 6	
5	
1 2 3 1 4	
10 10 5 2 5	
3	
1 6 4	
5 10 7	

Marks Distribution

Task	Marks
Task A	40%
Task B	40%
Task C	20%