

# Sequence

Supervin likes counting. In this problem, he invites you to count together

Supervin defines an **increasing sequence** is a non-empty sequence  $\{a_1, a_2, \dots, a_p\}$  such that for all  $i < p$ ,  $a_i < a_{i+1}$

Supervin defines a **decreasing sequence** is a non-empty sequence  $\{b_1, b_2, \dots, b_q\}$  such that for all  $i < q$ ,  $b_i > b_{i+1}$

Supervin has a sequence  $\{c_1, c_2, \dots, c_n\}$  and he wants to divide the sequence into two sequences. Supervin requires that one of the sequences is an increasing sequence, while the other one is a decreasing sequence. Supervin may rearrange the numbers in his sequence before dividing it.

You are given the initial sequence that Supervin has. You have to determine whether Supervin can divide the sequence according to his condition.

## Input

The first line consist of integer **T**, the number of cases (at most 20 each input file)

**T** cases follow. Each cases has :

Line 1 : Integer **N**, the number of integers in Supervin's sequence.

Line 2 : **N** space-separated integers indicating Supervin's sequence.

## Output

The output should contain **T** lines. The *i*-th line contains the output "YES" if Supervin can divide the *i*-th sequence as desired or "NO" if Supervin can't divide the *i*-th sequence as desired.

## Example

**Input:**

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

**Output:**

```
YES
```

**Explanation :**

For example, he can divide the sequence to :

Increasing sequence : {2,3,4}

Decreasing sequence : {5,2}

### **Constraints**

$1 \leq N \leq 50\,000$

Each number in Supervin's sequence is a non-negative integer less than 1 000 000 000