A. Submission Bait

time limit per test: 1 second
memory limit per test: 256 megabytes

Alice and Bob are playing a game in an array a of size n.

They take turns to do operations, with Alice starting first. The player who can not operate will lose. At first, a variable mx is set to 0.

In one operation, a player can do:

- Choose an index i ($1 \le i \le n$) such that $a_i \ge mx$ and set mx to a_i. Then, set a_i to 0.

Determine whether Alice has a winning strategy.

Input

The first line contains an integer t ($1 \le t \le 10^3$) — the number of test cases.

For each test case:

- The first line contains an integer n ($2 \le n \le 50$) — the size of the array.
- The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le n$) — the elements of the array.

Output

For each test case, if Alice has a winning strategy, output "YES". Otherwise, output "NO".

You can output the answer in any case (upper or lower). For example, the strings "yEs", "yes", "Yes", and "YES" will be recognized as positive responses.

Example

input Copy

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

output Copy

```
YES
NO
YES
NO
YES
```

Note

In the first test case, Alice can choose $i = 1$ since $a_1 = 2 \ge mx = 0$.

After Alice's operation, $a = [0, 1]$ and $mx = 2$. Bob can not do any operation. Alice wins.

In the second test case, Alice doesn't have a winning strategy.

For example, if Alice chooses $i = 1$, after Alice's operation: $a = [0, 1]$ and $mx = 1$. Then, Bob can choose $i = 2$ since $a_2 = 1 \ge mx = 1$. After Bob's operation: $a = [0, 0]$ and $mx = 1$. Alice can not do any operation. Bob wins.