Ordering Teams

Problem Statement

In ACM-ICPC contests, there are usually three people in a team. For each person in the team, you know their scores in three skills: hard work, intelligence, and persistence.

The goal is to determine whether it is possible to order these people (assign them numbers from 1 to 3) in such a way that for each $1 \leq i \leq 2$, the $(i+1)$-th person is **strictly better** than the i-th person.

Definition of Better: A person x is said to be better than another person y if:

- x doesn't score less than y in any of the skills.
- x scores more than y in at least one skill.

Input

- The first line of input contains an integer T denoting the number of test cases.
- Each test case consists of three lines. Each of these lines contains three space-separated integers s_1, s_2, and s_3 denoting the scores of one member of the team in each of the three skills, in the given order.

Output

For each test case, output a single line containing `yes` if such an ordering exists or `no` if it doesn't exist.

Constraints

- $1 \leq T \leq 1000$
- $1 \leq s_1, s_2, s_3 \leq 100$

Example

Input

```
3
1 2 3
2 3 4
2 3 5
1 2 3
2 3 4
2 3 4
5 6 5
1 2 3
2 3 4
```

Output

```
yes
no
yes
```

Explanation

- **Test Case 1:** We can order the team members as $(3, 2, 1)$. Person 3 is better than Person 2 because their scores in the first two skills are not less than Person 2's. In skill 3, Person 3 scores higher than Person 2. Similarly, Person 2 is better than Person 1, scoring higher in every skill.
- **Test Case 2:** No such ordering exists where each subsequent person is strictly better.
- **Test Case 3:** We can order the team members as $(1, 3, 2)$ to satisfy the conditions.