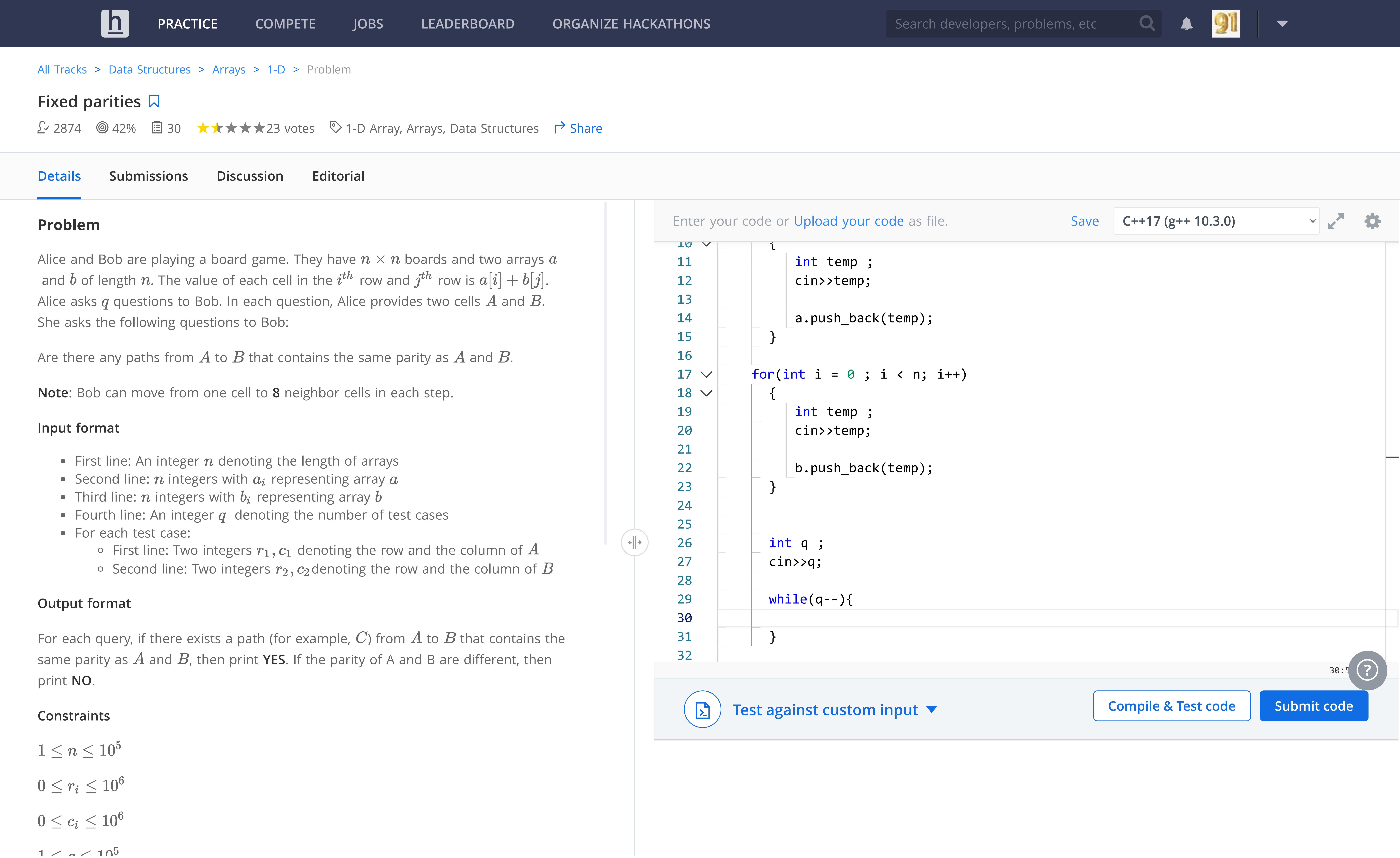
PRACTICE
COMPETE
JOBS
LEADERBOARD
ORGANIZE HACKATHONS
Search developers, problems, etc
All Tracks > Data Structures > Arrays > 1-D > Problem

# Fixed parities

2874
42%
30
23 votes
1-D Array, Arrays, Data Structures
Share
Details
Submissions
Discussion
Editorial

## Problem

Alice and Bob are playing a board game. They have $n \times n$ boards and two arrays $a$ and $b$ of length $n$. The value of each cell in the $i^{th}$ row and $j^{th}$ row is $a[i] + b[j]$. Alice asks $q$ questions to Bob. In each question, Alice provides two cells $A$ and $B$. She asks the following questions to Bob:

Are there any paths from $A$ to $B$ that contains the same parity as $A$ and $B$.

**Note**: Bob can move from one cell to **8** neighbor cells in each step.

### Input format

- First line: An integer $n$ denoting the length of arrays
- Second line: $n$ integers with $a_i$ representing array $a$
- Third line: $n$ integers with $b_i$ representing array $b$
- Fourth line: An integer $q$ denoting the number of test cases
- For each test case:
  - First line: Two integers $r_1, c_1$ denoting the row and the column of $A$
  - Second line: Two integers $r_2, c_2$ denoting the row and the column of $B$

### Output format

For each query, if there exists a path (for example, $C$) from $A$ to $B$ that contains the same parity as $A$ and $B$, then print **YES**. If the parity of A and B are different, then print **NO**.

### Constraints

$1 \leq n \leq 10^5$

$0 \leq r_i \leq 10^6$

$0 \leq c_i \leq 10^6$

$1 \leq q \leq 10^5$

```
    {
        int temp ;
        cin>>temp;

        a.push_back(temp);
    }

  for(int i = 0 ; i < n; i++)
    {
        int temp ;
        cin>>temp;

        b.push_back(temp);
    }


    int q ;
    cin>>q;

    while(q--){

    }

```