3189. Minimum Moves to Get a Peaceful Board Premium

Medium Topics Hint

Given a 2D array `rooks` of length `n`, where `rooks[i] = [xi, yi]` indicates the position of a rook on an `n x n` chess board. Your task is to move the rooks **1 cell** at a time vertically or horizontally (to an *adjacent* cell) such that the board becomes **peaceful**.

A board is **peaceful** if there is **exactly** one rook in each row and each column.

Return the **minimum** number of moves required to get a *peaceful board*.

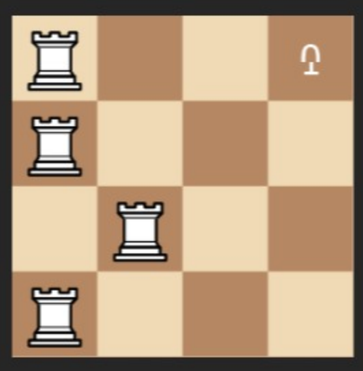
Note that **at no point** can there be two rooks in the same cell.

Example 1:

> **Input:** `rooks = [[0,0],[1,0],[1,1]]`
>
> **Output:** 3
>
> **Explanation:**

Example 2:

> **Input:** `rooks = [[0,0],[0,1],[0,2],[0,3]]`
>
> **Output:** 6
>
> **Explanation:**



Constraints:

- `1 <= n == rooks.length <= 500`
- `0 <= xi, yi <= n - 1`
- The input is generated such that there are no 2 rooks in the same cell.

Seen this question in a real interview before? 1/5

Yes No

Accepted **4.2K** | Submissions **5.5K** | Acceptance Rate **76.9%**

Topics

Array Greedy Sorting Counting Sort

Hint 1

Think of a greedy method.

Hint 2

First, distribute the rooks in individual rows.

Hint 3

You can do this by sorting all rooks by their rows. Then assign the first one to the first row, the second one to the second row, and so on.

Hint 4

After you've distributed rooks across all rows, now do the same for columns.

Hint 5

Sort rooks by their columns and then assign the first one to the first column and so on.

Similar Questions

Transform to Chessboard Hard

Most Stones Removed with Same Row or Column Medium

Discussion (11)