(/) Explore(/explore/) Problems(/problemset/all/) Interview New (/contest/) Contest Discuss(/discuss/) Store ☆ Premium (/subscribe?ref=nb_npl) 0 (/problems/number-of-good-paths/)

6292. Increment Submatrices by One

My Submissions (/contest/weekly-contest-328/problems/increment-submatrices-by-one/submissions/) Back to Contest (/contest/weekly-contest-328/)

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Medium

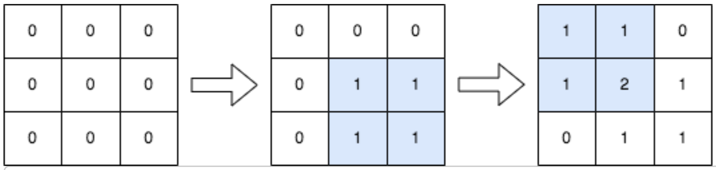
You are given a positive integer `n`, indicating that we initially have an `n x n` **0-indexed** integer matrix `mat` filled with zeroes.

You are also given a 2D integer array `query`. For each `query[i] = [`$row1_i$, $col1_i$, $row2_i$, $col2_i$`]`, you should do the following operation:

- Add `1` to **every element** in the submatrix with the **top left** corner ($row1_i$, $col1_i$) and the **bottom right** corner ($row2_i$, $col2_i$). That is, add `1` to `mat[x][y]` for for all $row1_i$ <= x <= $row2_i$ and $col1_i$ <= y <= $col2_i$.

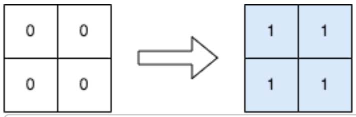
Return *the matrix* `mat` *after performing every query.*

Example 1:



```
Input: n = 3, queries = [[1,1,2,2],[0,0,1,1]]
Output: [[1,1,0],[1,2,1],[0,1,1]]
Explanation: The diagram above shows the initial matrix, the matrix after the first query, and the matrix after the second quer
- In the first query, we add 1 to every element in the submatrix with the top left corner (1, 1) and bottom right corner (2, 2)
- In the second query, we add 1 to every element in the submatrix with the top left corner (0, 0) and bottom right corner (1, 1
```

Example 2:



```
Input: n = 2, queries = [[0,0,1,1]]
Output: [[1,1],[1,1]]
Explanation: The diagram above shows the initial matrix and the matrix after the first query.
- In the first query we add 1 to every element in the matrix.
```

Constraints:

- `1 <= n <= 500`
- `1 <= queries.length <=` 10^4
- `0 <=` $row1_i$ `<=` $row2_i$ `< n`
- `0 <=` $col1_i$ `<=` $col2_i$ `< n`

JavaScript

```javascript
const initialize2DArray = (n, m) => [...Array(n)].map(() => Array(m).fill(0));

const rangeAddQueries = (n, q) => {
    let g = initialize2DArray(n, n);
    for (const [x1, y1, x2, y2] of q)
        for (let r = x1; r <= x2; r++)
            for (let c = y1; c <= y2; c++) g[r][c]++;
    return g;
};
```

□ **Custom Testcase** Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/878336914/) More Details > (/submissions/detail/878336914/)

Share your acceptance!

Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Online Interview (/interview/) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)

United States (/region)