5845. Last Day Where You Can Still Cross

My Submissions (/contest/weekly-contest-254/problems/last-day-where-you-can-still-cross/submissions/)

Back to Contest (/contest/weekly-contest-254/)

User Accepted:	0
User Tried:	2
Total Accepted:	0
Total Submissions:	2
Difficulty:	Hard

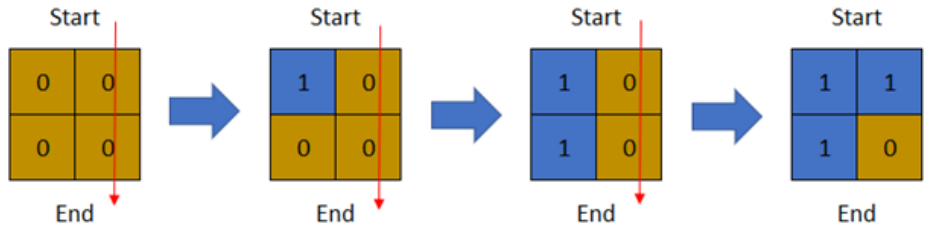
There is a **1-based** binary matrix where `0` represents land and `1` represents water. You are given integers `row` and `col` representing the number of rows and columns in the matrix, respectively.

Initially on day `0`, the **entire** matrix is **land**. However, each day a new cell becomes flooded with **water**. You are given a **1-based** 2D array `cells`, where `cells[i] = [r_i, c_i]` represents that on the `i^th` day, the cell on the `r_i^th` row and `c_i^th` column (**1-based** coordinates) will be covered with **water** (i.e., changed to `1`).

You want to find the **last** day that it is possible to walk from the **top** to the **bottom** by only walking on land cells. You can start from **any** cell in the top row and end at **any** cell in the bottom row. You can only travel in the **four** cardinal directions (left, right, up, and down).

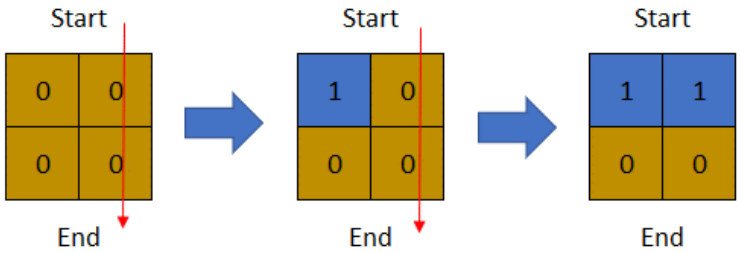
Return *the **last** day where it is possible to walk from the **top** to the **bottom** by only walking on land cells*.

Example 1:



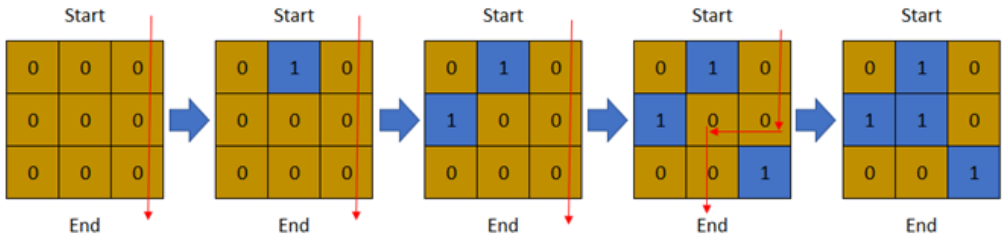

```
Input: row = 2, col = 2, cells = [[1,1],[2,1],[1,2],[2,2]]
Output: 2
Explanation: The above image depicts how the matrix changes each day starting from day 0.
The last day where it is possible to cross from top to bottom is on day 2.
```

Example 2:




```
Input: row = 2, col = 2, cells = [[1,1],[1,2],[2,1],[2,2]]
Output: 1
Explanation: The above image depicts how the matrix changes each day starting from day 0.
The last day where it is possible to cross from top to bottom is on day 1.
```

Example 3:



Input: row = 3, col = 3, cells = [[1,2],[2,1],[3,3],[2,2],[1,1],[1,3],[2,3],[3,2],[3,1]]
Output: 3
Explanation: The above image depicts how the matrix changes each day starting from day 0.
The last day where it is possible to cross from top to bottom is on day 3.

Constraints:

- $2 <= row, col <= 2 * 10^4$
- $4 <= row * col <= 2 * 10^4$
- cells.length == row * col
- $1 <= r_i <= row$
- $1 <= c_i <= col$
- All the values of cells are **unique**.

JavaScript

```
const initialize2DArrayNew = (n, m) => { let data = []; for (let i = 0; i < n; i++) { let tmp = Array(m).fill(1); data.push(tmp); } return data; };

const dx = [-1, 1, 0, 0], dy = [0, 0, -1, 1];
let grid, n, m;
const latestDayToCross = (row, col, cells) => {
    n = row, m = col;
    let low = 0, high = n * m;
    while (low < high) {
        let mid = low + high + 1 >> 1;
        let visit = initialize2DArrayNew(n, m);
        bfs(visit, mid, cells) ? low = mid : high = mid - 1;
    }
    return low;
};

const bfs = (visit, mid, cells) => {
    for (let i = 0; i < mid; i++) {
        let [x, y] = cells[i];
        visit[x - 1][y - 1] = 0;
    }
    let q = [];
    for (let j = 0; j < m; j++) { // first row start
        if (visit[0][j]) {
            visit[0][j] = 2;
            q.push([0, j]);
        }
    }
    while (q.length) {
        let [x, y] = q.shift();
        for (let k = 0; k < 4; k++) {
            let nx = x + dx[k], ny = y + dy[k];
            if (nx >= 0 && nx < n && ny >= 0 && ny < m) {
                if (visit[nx][ny] == 1) {
                    visit[nx][ny] = 2;
                    q.push([nx, ny]);
                }
            }
```

```
        }
    }
    let ok = false;
    for (let j = 0; j < m; j++) {
        ok |= visit[n - 1][j] == 2;
    }
    return ok;
};
```

Custom Testcase | Use Example Testcases

Run | Submit

Submission Result: Accepted (/submissions/detail/538723214/) | More Details > (/submissions/detail/538723214/)

Share your acceptance!

1