6376. Row With Maximum Ones

My Submissions (/contest/weekly-contest-341/problems/row-with-maximum-ones/submissions/) Back to Contest (/contest/weekly-contest-341/)

Given a `m x n` binary matrix `mat`, find the **0-indexed** position of the row that contains the **maximum** count of **ones,** and the number of ones in that row.

In case there are multiple rows that have the maximum count of ones, the row with the **smallest row number** should be selected.

Return *an array containing the index of the row, and the number of ones in it.*

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

Example 1:

```
Input: mat = [[0,1],[1,0]]
Output: [0,1]
Explanation: Both rows have the same number of 1's. So we return the index of the smaller row, 0, and the maximum count of ones
. So, the answer is [0,1].
```

Example 2:

```
Input: mat = [[0,0,0],[0,1,1]]
Output: [1,2]
Explanation: The row indexed 1 has the maximum count of ones (2). So we return its index, 1
, and the count. So, the answer is [1,2].
```

Example 3:

```
Input: mat = [[0,0],[1,1],[0,0]]
Output: [1,2]
Explanation: The row indexed 1 has the maximum count of ones (2). So the answer is [1,2].
```

Constraints:

- `m == mat.length`
- `n == mat[i].length`
- `1 <= m, n <= 100`
- `mat[i][j]` is either `0` or `1`.

JavaScript

```
const rowAndMaximumOnes = (g) => g.map((a, i) => [i, a.filter(x => x == 1).length]).sort((x, y) => y[1] - x[1] || x[0] -
y[0])[0];
```

Custom Testcase Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/934441934/) More Details (/submissions/detail/934441934/)

Share your acceptance!