LeetCode Explore (Day 4) Problems Interview (New) Contest Discuss Store

Description | Solution | Discuss (589) | Submissions

Python3 | Autocomplete

1428. Leftmost Column with at Least a One

Medium 574 75 Add to List Share

*(This problem is an **interactive problem**.)*

A **row-sorted binary matrix** means that all elements are `0` or `1` and each row of the matrix is sorted in non-decreasing order.

Given a **row-sorted binary matrix** `binaryMatrix`, return *the index (0-indexed) of the **leftmost column** with a 1 in it*. If such an index does not exist, return `-1`.

You can't access the Binary Matrix directly. You may only access the matrix using a `BinaryMatrix` interface:

- `BinaryMatrix.get(row, col)` returns the element of the matrix at index `(row, col)` (0-indexed).
- `BinaryMatrix.dimensions()` returns the dimensions of the matrix as a list of 2 elements `[rows, cols]`, which means the matrix is `rows x cols`.

Submissions making more than `1000` calls to `BinaryMatrix.get` will be judged *Wrong Answer*. Also, any solutions that attempt to circumvent the judge will result in disqualification.

For custom testing purposes, the input will be the entire binary matrix `mat`. You will not have access to the binary matrix directly.

Example 1:

0	0
1	1

```
Input: mat = [[0,0],[1,1]]
Output: 0
```

Example 2:

0	**0**
0	**1**

```
Input: mat = [[0,0],[0,1]]
Output: 1
```

Example 3:

0	0
0	0

```
Input: mat = [[0,0],[0,0]]
Output: -1
```

Example 4:

0	**0**	0	1
0	**0**	1	1
0	**1**	1	1

```
Input: mat = [[0,0,0,1],[0,0,1,1],[0,1,1,1]]
Output: 1
```

Constraints:

- `rows == mat.length`
- `cols == mat[i].length`
- `1 <= rows, cols <= 100`
- `mat[i][j]` is either `0` or `1`.
- `mat[i]` is sorted in non-decreasing order.

Accepted 97,813 Submissions 195,771

Seen this question in a real interview before? Yes No

Companies *i*

0 ~ 6 months 6 months ~ 1 year 1 year ~ 2 years

Facebook | 62

Related Topics

Show Hint 1

Show Hint 2

```
# """
# This is BinaryMatrix's API interface.
# You should not implement it, or speculate about its implementation
# """
#class BinaryMatrix(object):
#    def get(self, row: int, col: int) -> int:
#    def dimensions(self) -> list[]:

class Solution:
    def leftMostColumnWithOne(self, binaryMatrix: 'BinaryMatrix') -> int:
        
```

Problems | Pick One | Prev | 1/1 | Next | Console | Contribute *i* | Run Code | Submit