Unknown Title

Description

Description

Note

Note

Editorial

Editorial

Solutions

Solutions

Submissions

Submissions

Code

Code

☑

Testcase

Testcase

>_

Test Result

Test Result

45. Jump Game II

Medium

Topics

Companies

You are given a **0-indexed** array of integers `nums` of length `n`. You are initially positioned at `nums[0]`.

Each element `nums[i]` represents the maximum length of a forward jump from index `i`. In other words, if you are at `nums[i]`, you can jump to any `nums[i + j]` where:

- `0 <= j <= nums[i]` and
- `i + j < n`

Return *the minimum number of jumps to reach* `nums[n - 1]`. The test cases are generated such that you can reach `nums[n - 1]`.

Example 1:

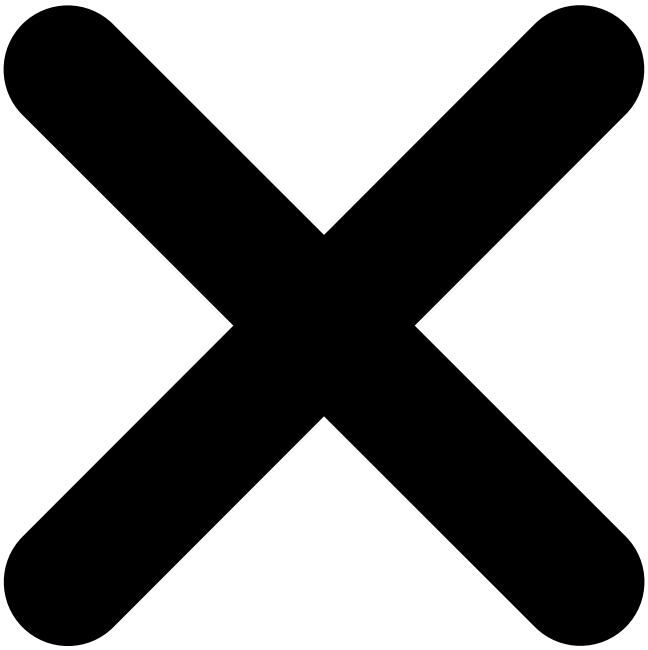
```
Input: nums = [2,3,1,1,4]
Output: 2
Explanation: The minimum number of jumps to reach the last index is 2. Jump 1 step from index 0 to 1, then 3 steps to the last index.
```

Example 2:

```
Input: nums = [2,3,0,1,4]
Output: 2
```

Constraints:

- `1 <= nums.length <= 10^4`
- `0 <= nums[i] <= 1000`
- It's guaranteed that you can reach `nums[n - 1]`.



Seen this question in a real interview before?

1/5

Yes

No

Accepted

1.4M

Submissions

3.5M

Acceptance Rate

40.8%

Companies

Discussion (155)

‹/›
@

💡 Discussion Rules

1. Please don't post **any solutions** in this discussion.

2. The problem discussion is for asking questions about the problem or for sharing tips - anything except for solutions.

3. If you'd like to share your solution for feedback and ideas, please head to the solutions tab and post it there.

No comments yet.

1

2

3

4

5

```
class Solution {
    public int jump(int[] nums) {
```

```
    }
}
```

Saved

Ln 1, Col 1

nums =

[2,3,1,1,4]

1

[2,3,1,1,4]

</>
Source

FindHeaderBarSize

FindTabBarSize

FindBorderBarSize