251. Flatten 2D Vector

Medium 652 360 Add to List Share

Design an iterator to flatten a 2D vector. It should support the `next` and `hasNext` operations.

Implement the `Vector2D` class:

- `Vector2D(int[][] vec)` initializes the object with the 2D vector `vec`.
- `next()` returns the next element from the 2D vector and moves the pointer one step forward. You may as calls to `next` are valid.
- `hasNext()` returns `true` if there are still some elements in the vector, and `false` otherwise.

Example 1:

```
Input
["Vector2D", "next", "next", "next", "hasNext", "hasNext", "next", "hasNext"]
[[[[1, 2], [3], [4]]], [], [], [], [], [], [], []]
Output
[null, 1, 2, 3, true, true, 4, false]

Explanation
Vector2D vector2D = new Vector2D([[1, 2], [3], [4]]);
vector2D.next();    // return 1
vector2D.next();    // return 2
vector2D.next();    // return 3
vector2D.hasNext(); // return True
vector2D.hasNext(); // return True
vector2D.next();    // return 4
vector2D.hasNext(); // return False
```

Constraints:

- `0 <= vec.length <= 200`
- `0 <= vec[i].length <= 500`
- `-500 <= vec[i][j] <= 500`
- At most 10^5 calls will be made to `next` and `hasNext`.