Description | Accepted | Editorial | Submissions | Solutions

1121. Divide Array Into Increasing Sequences

Premium

Solved

Hard | Topics | Companies | Hint

Given an integer array `nums` sorted in non-decreasing order and an integer `k`, return `true` *if this array can be divided into one or more disjoint increasing subsequences of length at least* `k`, *or* `false` *otherwise.*

Example 1:

```
Input: nums = [1,2,2,3,3,4,4], k = 3
Output: true
Explanation: The array can be divided into two subsequences [1,2,3,4] and [2,3,4] with lengths at least 3 each.
```

Example 2:

```
Input: nums = [5,6,6,7,8], k = 3
Output: false
Explanation: There is no way to divide the array using the conditions required.
```

Constraints:

- `1 <= k <= nums.length <= 10^5`
- `1 <= nums[i] <= 10^5`
- `nums` is sorted in non-decreasing order.

Seen this question in a real interview before? 1/5

Yes | No

Accepted 4.8K | Submissions 7.8K | Acceptance Rate 61.3%

Topics

Companies

Hint 1

Think in the frequency of the numbers and how this affects the number of sequences needed.

Hint 2

What is the minimum number of sequences we need to form? Considering frequency of the numbers.

Hint 3

Hint 4

Hint 5

Discussion (2)

121 | 2

Code

Python3 • Auto

```python
class Solution:
    def canDivideIntoSubsequences(self, nums: List[int], k: int) -> bool:

        dictx = collections.Counter(nums)

        max_val = max(dictx.values())

        return max_val*k <= len(nums)

```

Saved

Ln 9, Col 9

Testcase | Test Result

Accepted Runtime: 34 ms

Case 1 • Case 2

Input

nums =
[1,2,2,3,3,4,4]

k =
3

Stdout

```
Counter({2: 2, 3: 2, 4: 2, 1: 1})
```

Output