5723. Finding the Users Active Minutes

My Submissions (/contest/weekly-contest-235/problems/finding-the-users-active-minutes/submissions/)

Back to Contest (/contest/weekly-contest-235/)

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

You are given the logs for users' actions on LeetCode, and an integer `k`. The logs are represented by a 2D integer array `logs` where each `logs[i] = [ID_i, time_i]` indicates that the user with `ID_i` performed an action at the minute `time_i`.

Multiple users can perform actions simultaneously, and a single user can perform **multiple actions** in the same minute.

The **user active minutes (UAM)** for a given user is defined as the **number of unique minutes** in which the user performed an action on LeetCode. A minute can only be counted once, even if multiple actions occur during it.

You are to calculate a **1-indexed** array `answer` of size `k` such that, for each `j` (`1 <= j <= k`), `answer[j]` is the **number of users** whose **UAM** equals `j`.

Return *the array* `answer` *as described above.*

Example 1:

```
Input: logs = [[0,5],[1,2],[0,2],[0,5],[1,3]], k = 5
Output: [0,2,0,0,0]
Explanation:
The user with ID=0 performed actions at minutes 5, 2, and 5 again. Hence, they have a UAM of 2 (minute
The user with ID=1 performed actions at minutes 2 and 3. Hence, they have a UAM of 2.
Since both users have a UAM of 2, answer[2] is 2, and the remaining answer[j] values are 0.
```

Example 2:

```
Input: logs = [[1,1],[2,2],[2,3]], k = 4
Output: [1,1,0,0]
Explanation:
The user with ID=1 performed a single action at minute 1. Hence, they have a UAM of 1.
The user with ID=2 performed actions at minutes 2 and 3. Hence, they have a UAM of 2.
There is one user with a UAM of 1 and one with a UAM of 2.
Hence, answer[1] = 1, answer[2] = 1, and the remaining values are 0.
```

Constraints:

- `1 <= logs.length <= 10^4`
- `0 <= ID_i <= 10^9`
- `1 <= time_i <= 10^5`
- `k` is in the range `[The maximum UAM for a user, 10^5]`.

JavaScript

```
1 ▾ const findingUsersActiveMinutes = (logs, k) => {
```

```
    let m = new Map();
    for (const e of logs) {
        if (!m.has(e[0])) m.set(e[0], new Set());
        m.get(e[0]).add(e[1]);
    }
    let uam = new Map();
    for (const [k, v] of m) {
        uam.set(k, v.size);
    }
    let res = Array(k).fill(0);
    for (const [k, v] of uam) {
        res[v - 1]++;
    }
    return res;
};
```

Custom Testcase | Use Example Testcases

Run | Submit

Submission Result: Accepted (/submissions/detail/476166883/)

More Details > (/submissions/detail/476166883/)

Share your acceptance!