# 5539. Sort Array by Increasing Frequency

My Submissions (/contest/biweekly-contest-38/problems/sort-array-by-increasing-frequency/submissions/)

Back to Contest (/contest/biweekly-contest-38/)

Given an array of integers `nums` , sort the array in **increasing** order based on the frequency of the values. If multiple values have the same frequency, sort them in **decreasing** order.

Return the *sorted array*.

| | |
|---|---|
| User Accepted: | 1587 |
| User Tried: | 1909 |
| Total Accepted: | 1600 |
| Total Submissions: | 2280 |
| Difficulty: | Easy |

**Example 1:**

```
Input: nums = [1,1,2,2,2,3]
Output: [3,1,1,2,2,2]
Explanation: '3' has a frequency of 1, '1' has a frequency of 2, and '2' has a frequency o
```

**Example 2:**

```
Input: nums = [2,3,1,3,2]
Output: [1,3,3,2,2]
Explanation: '2' and '3' both have a frequency of 2, so they are sorted in decreasing order.
```

**Example 3:**

```
Input: nums = [-1,1,-6,4,5,-6,1,4,1]
Output: [5,-1,4,4,-6,-6,1,1,1]
```

**Constraints:**

- `1 <= nums.length <= 100`
- `-100 <= nums[i] <= 100`

JavaScript

```
/**
 * @param {number[]} nums
 * @return {number[]}
 */
const frequencySort = (nums) => {
    let record = getRecord(nums);
    nums.sort((a, b) => {
        if (record.get(a) == record.get(b)) return b - a;
        return record.get(a) - record.get(b);
    });
    return nums;
};

const getRecord = (s) => {
    let map = new Map();
    for (const i of s) {
        if (map.has(i)) {
            map.set(i, map.get(i) + 1);
        } else {
            map.set(i, 1);
        }
    }
    return map;
};
```

Custom Testcase Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/415232300/) More Details (/submissions/detail/415232300/)

Share your acceptance!

1