

5217. Sort the Jumbled Numbers

My Submissions (/contest/biweekly-contest-73/problems/sort-the-jumbled-numbers/submissions/)

Back to Contest (/contest/biweekly-contest-73/)

You are given a **0-indexed** integer array `mapping` which represents the mapping rule of a shuffled decimal system. `mapping[i] = j` means digit `i` should be mapped to digit `j` in this system.

The **mapped value** of an integer is the new integer obtained by replacing each occurrence of digit `i` in the integer with `mapping[i]` for all $0 \leq i \leq 9$.

You are also given another integer array `nums`. Return *the array* `nums` sorted in **non-decreasing order based on the mapped values** of its elements.

Notes:

- Elements with the same mapped values should appear in the **same relative order** as in the input.
- The elements of `nums` should only be sorted based on their mapped values and **not be replaced** by them.

User Accepted:

0

User Tried:

0

Total Accepted:

0

Total Submissions:

0

Difficulty:

Medium

Example 1:

Input: mapping = [8,9,4,0,2,1,3,5,7,6], nums = [991,338,38]

Output: [338,38,991]

Explanation:

Map the number 991 as follows:
1. mapping[9] = 6, so all occurrences of the digit 9 will become 6.
2. mapping[1] = 9, so all occurrences of the digit 1 will become 9.
Therefore, the mapped value of 991 is 669.
338 maps to 007, or 7 after removing the leading zeros.
38 maps to 07, which is also 7 after removing leading zeros.
Since 338 and 38 share the same mapped value, they should remain in the same relative order, so 338 comes before 38.
Thus, the sorted array is [338,38,991].

Example 2:

Input: mapping = [0,1,2,3,4,5,6,7,8,9], nums = [789,456,123]

Output: [123,456,789]

Explanation: 789 maps to 789, 456 maps to 456, and 123 maps to 123. Thus, the sorted array is [123,456,789].

Constraints:

- mapping.length == 10
- 0 <= mapping[i] <= 9
- All the values of mapping[i] are **unique**.
- 1 <= nums.length <= 3 * 10⁴
- 0 <= nums[i] < 10⁹

JavaScript

```
1 const sortJumbled = (m, a) => {
2   let res = [];
3   for (let i = 0; i < a.length; i++) {
4     let s = a[i] + '', t = '';
5     for (const c of s) t += m[c - '0'];
6     res.push([a[i], t - '0', i]);
7   }
8   res.sort((x, y) => {
9     if (x[1] != y[1]) return x[1] - y[1];
10    return x[2] - y[2];
11  });
12   return res.map(x => x[0]);
13 };
```

☐ Custom Testcase

Use Example Testcases

Run

Submit

Submission Result: **Accepted** (/submissions/detail/653832572/) ⓘ More Details > (/submissions/detail/653832572/)

Share your acceptance!