Day 15
(/)Explore   Problems(/problemset/all/)   Interview   New   (/contest/) Contest   Discuss(/discuss/)   🚀May LeetCoding Challenge(/discuss/general-Storediscussion/655704/)   ☆ Premium (/subscribe?ref=nb_npl)

5742. Sorting the Sentence

My Submissions (/contest/biweekly-contest-52/problems/sorting-the-sentence/submissions/)

Back to Contest (/contest/biweekly-contest-52/)

A **sentence** is a list of words that are separated by a single space with no leading or trailing spaces. Each word consists of lowercase and uppercase English letters.

A sentence can be **shuffled** by appending the **1-indexed word position** to each word then rearranging the words in the sentence.

- For example, the sentence `"This is a sentence"` can be shuffled as `"sentence4 a3 is2 This1"` or `"is2 sentence4 This1 a3"`.

Given a **shuffled sentence** `s` containing no more than `9` words, reconstruct and return *the original sentence*.

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

Example 1:

```
Input: s = "is2 sentence4 This1 a3"
Output: "This is a sentence"
Explanation: Sort the words in s to their original positions "This1 is2 a3 sentence4", then remove the numbers.
```

Example 2:

```
Input: s = "Myself2 Me1 I4 and3"
Output: "Me Myself and I"
Explanation: Sort the words in s to their original positions "Me1 Myself2 and3 I4", then remove the numbers.
```

Constraints:

- `2 <= s.length <= 200`
- `s` consists of lowercase and uppercase English letters, spaces, and digits from `1` to `9`.
- The number of words in `s` is between `1` and `9`.
- The words in `s` are separated by a single space.
- `s` contains no leading or trailing spaces.

JavaScript ▼

```javascript
1 ▾ const sortSentence = (s) => {
2       let a = s.split(" ");
3       let res = [];
4 ▾     for (const e of a) {
5           let n = e.length;
6 ▾         for (let i = n - 1; ~i; i--) {
7 ▾             if (isLetter(e[i])) {
8                   res.push([e.slice(0, i + 1), Number(e.slice(i + 1))]);
9                   break;
10              }
11          }
12      }
13      res.sort((x, y) => x[1] - y[1]);
14      res = res.map(x => x[0]);
15      return res.join(" ");
16 };
```

```
17
18 ▾ const isLetter = (c) => {
19       return (c.charCodeAt() >= 65 && c.charCodeAt() <= 90) || (c.charCodeAt() >= 97 && c.charCodeAt() <=
         122);
20   };
```

☐ **Custom Testcase**   [Use Example Testcases]

[▶ Run]   [☁ Submit]

Submission Result: Accepted (/submissions/detail/493485900/) ❓   [More Details ❯ (/submissions/detail/493485900/)]

Share your acceptance!

Copyright © 2021 LeetCode

Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)

🇺🇸 United States (/region)