--(/) Explore Problems(/problemset/all/) Mock(/interview/)

Contest Discuss(/discuss/)

₩₩wek#ng, apply today!
Storedata=eyJ1cmwiOiAiaHR0cHM6Ly9sZWV0Y29kZS5jb20vam9icy8_cmVmPX

5714. Evaluate the Bracket Pairs of a String

My Submissions (/contest/weekly-contest-234/problems/evaluate-the-bracket-pairs-of-a-string/submissions/)

Back to Contest (/contest/weekly-contest-234/)

0

0

0

0

Medium

User Accepted:

Total Accepted:

Total Submissions:

User Tried:

Difficulty:

You are given a string s that contains some bracket pairs, with each pair containing a non-empty key.

For example, in the string "(name)is(age)yearsold", there are two bracket pairs that contain the keys "name" and "age".

You know the values of a wide range of keys. This is represented by a 2D string array knowledge where each knowledge $[i] = [key_i, value_i]$ indicates that key key_i has a value of $value_i$.

You are tasked to evaluate all of the bracket pairs. When you evaluate a bracket pair that contains some key key_1 , you will:

- Replace key_i and the bracket pair with the key's corresponding value_i.
- If you do not know the value of the key, you will replace key₁ and the bracket pair with a question mark "?" (without the quotation marks).

Each key will appear at most once in your knowledge . There will not be any nested brackets in s .

Return the resulting string after evaluating all of the bracket pairs.

Example 1:

```
Input: s = "(name)is(age)yearsold", knowledge = [["name","bob"],["age","two"]]
Output: "bobistwoyearsold"
Explanation:
The key "name" has a value of "bob", so replace "(name)" with "bob".
The key "age" has a value of "two", so replace "(age)" with "two".
```

Example 2:

```
Input: s = "hi(name)", knowledge = [["a","b"]]
Output: "hi?"
Explanation: As you do not know the value of the key "name", replace "(name)" with "?".
```

Example 3:

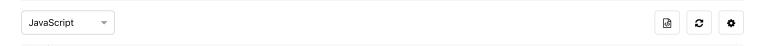
```
Input: s = "(a)(a)(a)aaa", knowledge = [["a","yes"]]
Output: "yesyesyesaaa"
Explanation: The same key can appear multiple times.
The key "a" has a value of "yes", so replace all occurrences of "(a)" with "yes".
Notice that the "a"s not in a bracket pair are not evaluated.
```

Example 4:

```
Input: s = "(a)(b)", knowledge = [["a","b"],["b","a"]]
Output: "ba"
```

Constraints:

- 1 <= s.length <= 10^5
- 0 <= knowledge.length <= 10^5
- knowledge[i].length == 2
- 1 <= key_i .length, $value_i$.length <= 10
- s consists of lowercase English letters and round brackets '(' and ')'.
- Every open bracket '(' in s will have a corresponding close bracket ')'.
- The key in each bracket pair of s will be non-empty.
- There will not be any nested bracket pairs in s .
- key_i and value_i consist of lowercase English letters.
- Each key_i in knowledge is unique.



```
1 \cdot | const evaluate = (s, knowledge) => {
 2
         let m = new Map();
         for (const e of knowledge) {
3 •
 4
             m.set(e[0], e[1]);
 5
 6
         let st = '';
 7 •
         for (const c of s) {
             if (st.length != 0) {
   if (c == ')') {
 8 ▼
 9 ▼
                      let leftIdx = st.lastIndexOf('(');
10
                      let ke = st.slice(leftIdx + 1);
11
12
                      st = st.slice(0, leftIdx);
13 ▼
                      if (m.has(ke)) {
14
                           st += m.get(ke);
15 ▼
                      } else {
16
                           st += '?';
17
                      }
18 ▼
                  } else {
19
                      st += c;
20
             } else {
21 •
22
                  st += c;
23
24
        }
25
         return st;
26
    };
```

☐ Custom Testcase

United States (/region)

Use Example Testcases

Submission Result: Accepted (/submissions/detail/473292015/)

More Details > (/submissions/detail/473292015/)

Share your acceptance!

Copyright © 2021 LeetCode Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)

https://leetcode.com/contest/weekly-contest-234/problems/evaluate-the-bracket-pairs-of-a-string/