



890. Find and Replace Pattern

Notes

[Description \(/problems/find-and-replace-pattern/description/\)](/problems/find-and-replace-pattern/description/)[Hints \(/problems/find-and-replace-pattern/hints/\)](/problems/find-and-replace-pattern/hints/)[Submissions \(/problems/find-and-replace-pattern/submissions/\)](/problems/find-and-replace-pattern/submissions/)[Pick One \(/problems/random-one-question/\)](/problems/random-one-question/)

You have a list of `words` and a `pattern`, and you want to know which words in `words` matches the pattern.

A word matches the pattern if there exists a permutation of letters `p` so that after replacing every letter `x` in the pattern with `p(x)`, we get the desired word.

(Recall that a permutation of letters is a bijection from letters to letters: every letter maps to another letter, and no two letters map to the same letter.)

Return a list of the words in `words` that match the given pattern.

You may return the answer in any order.

Example 1:

Input: `words = ["abc","deq","mee","aqq","dkd","ccc"], pattern = "abb"`
Output: `["mee","aqq"]`
Explanation: "mee" matches the pattern because there is a permutation `{a -> m, b -> e, ...}`.
"ccc" does not match the pattern because `{a -> c, b -> c, ...}` is not a permutation, since a and b map to the same letter.

Note:

- `1 <= words.length <= 50`
- `1 <= pattern.length = words[i].length <= 20`

Seen this question in a real interview before?

Companies

[Amazon !\[\]\(bff896c19919791b89ab521f039b410a_img.jpg\) \(/company/amazon\)](/company/amazon/)

Related Topics

[String \(/tag/string\)](/tag/string/)

Java



```
1 class Solution {
2     public List<String> findAndReplacePattern(String[] words, String pattern) {
3         List<String> result = new ArrayList<>();
4         for (String word : words) {
5             if (matches(word, pattern)) {
6                 result.add(word);
7             }
8         }
9         return result;
10    }
11
12    private boolean matches(String word, String pattern) {
13        if (word.length() != pattern.length()) {
14            return false;
15        }
16        Map<Character, Character> map = new HashMap<>();
17        for (int i = 0; i < word.length(); i++) {
18            char w = word.charAt(i);
19            char p = pattern.charAt(i);
20            if (map.containsKey(w)) {
21                if (map.get(w) != p) return false;
22            } else {
23                map.put(w, p);
24            }
25        }
26        return true;
27    }
28 }
```

```
15     }
16
17     Map<Character, Character> w2p = new HashMap<>();
18     for (int i=0; i<word.length(); i++) {
19         Character w = word.charAt(i);
20         Character p = pattern.charAt(i);
21         Character mapped = w2p.get(w);
22         if (mapped == null) {
23             w2p.put(w, p);
24         } else if (!mapped.equals(p)) {
25             return false;
26         }
27     }
28
29     if (w2p.size() != new HashSet<>(w2p.values()).size()) {
30         return false;
31     }
32
33     return true;
34 }
35 }
```

Notes

☐ Custom Testcase ([Contribute](#))[Run](#)[Submit](#)Submission Result: **Accepted** (</submissions/detail/172175876/>) ?[More Details >](/submissions/detail/172175876/)

Next challenges:

[Next Closest Time \(/problems/next-closest-time/\)](/problems/next-closest-time/)[Count Binary Substrings \(/problems/count-binary-substrings/\)](/problems/count-binary-substrings/)[Masking Personal Information \(/problems/masking-personal-information/\)](/problems/masking-personal-information/)

Share your acceptance!

Check out our solution!

[Reveal Solution > \(/articles/find-and-replace-pattern/\)](/articles/find-and-replace-pattern/)

Copyright © 2018 LeetCode

[Contact Us \(/support/\)](/support/) | [Frequently Asked Questions \(/faq/\)](/faq/) | [Terms of Service \(/terms/\)](/terms/) | [Privacy Policy \(/privacy/\)](/privacy/) [United States \(/region/\)](/region/)