LeetCode Explore Problems Interview ^{New} Contest Discuss LeetCode is hiring! Apply NOW. Store ⭐ Premium 🔔 5 🔥 11 👤

📄 Description   ⚗ Solution   💬 Discuss (999+)   🕐 Submissions

i C++

890. Find and Replace Pattern

Medium 👍 3203 👎 151 ♡ Add to List ⤷ Share

Given a list of strings `words` and a string `pattern`, return *a list of* `words[i]` *that match* `pattern`. You may return the answer in **any order**.

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.

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.
```

Example 2:

```
Input: words = ["a","b","c"], pattern = "a"
Output: ["a","b","c"]
```

Constraints:

- `1 <= pattern.length <= 20`
- `1 <= words.length <= 50`
- `words[i].length == pattern.length`
- `pattern` and `words[i]` are lowercase English letters.

Accepted 152,266   Submissions 195,445

Seen this question in a real interview before? Yes No

Companies 🔒 ⌄

Related Topics ⌄

Problems   Pick One   ‹ Prev 890/2356 Next ›

```cpp
class Solution {
public:
    vector<string>
findAndReplacePattern(vector<string>& words, string pattern) {
        vector<string> ans;
        for(int i=0;
i<words.size(); i++){
            map<char, char> m1;
            map<char, char> m2;
            int flag = 1;

            for(int j=0;
j<words[i].size(); j++){
                auto it =
m1.find(pattern[j]);
                auto it2 =
m2.find(words[i][j]);

                if(it ==
m1.end()){

 m1.insert({pattern[j], words[i][j]});
                }
                else{
```

Autocomplete

Your previous code was restored from your local storage.

Testcase   Run Code Result   Debugger 🔒 ⌄

Accepted Runtime: 2 ms

Your input
```
["abc","deq","mee","aqq","d
"abb"
```

Output
```
["mee","aqq"]
```
Diff

Expected
```
["mee","aqq"]
```

Console ⌃   Use Example Testcases

▶ Run Code ⌃   Submit