Problem 966: Vowel Spellchecker

Problem Information

Difficulty: Medium
Acceptance Rate: 61.45%
Paid Only: No
Tags: Array, Hash Table, String

Problem Description

Given a `wordlist`, we want to implement a spellchecker that converts a query word into a correct word.

For a given `query` word, the spell checker handles two categories of spelling mistakes:

* Capitalization: If the query matches a word in the wordlist (**case-insensitive**), then the query word is returned with the same case as the case in the wordlist. * Example: `wordlist = ["yellow"]`, `query = "YellOw"`: `correct = "yellow"` * Example: `wordlist = ["Yellow"]`, `query = "yellow"`: `correct = "Yellow"` * Example: `wordlist = ["yellow"]`, `query = "yellow"`: `correct = "yellow"` * Vowel Errors: If after replacing the vowels `('a', 'e', 'i', 'o', 'u')` of the query word with any vowel individually, it matches a word in the wordlist (**case-insensitive**), then the query word is returned with the same case as the match in the wordlist. * Example: `wordlist = ["YellOw"]`, `query = "yollow"`: `correct = "YellOw"` * Example: `wordlist = ["YellOw"]`, `query = "yeellow"`: `correct = ""` (no match) * Example: `wordlist = ["YellOw"]`, `query = "yllw"`: `correct = ""` (no match)

In addition, the spell checker operates under the following precedence rules:

* When the query exactly matches a word in the wordlist (**case-sensitive**), you should return the same word back. * When the query matches a word up to capitalization, you should return the first such match in the wordlist. * When the query matches a word up to vowel errors, you should return the first such match in the wordlist. * If the query has no matches in the wordlist, you should return the empty string.

Given some `queries`, return a list of words `answer`, where `answer[i]` is the correct word for `query = queries[i]`.

Example 1:

Input: wordlist = ["KiTe","kite","hare","Hare"], queries = ["kite","Kite","KiTe","Hare","HARE","Hear","hear","keti","keet","keto"] **Output:** ["kite","KiTe","KiTe","Hare","hare","","","KiTe","","KiTe"]

Example 2:

Input: wordlist = ["yellow"], queries = ["YellOw"] **Output:** ["yellow"]

Constraints:

* `1 <= wordlist.length, queries.length <= 5000` * `1 <= wordlist[i].length, queries[i].length <= 7` * `wordlist[i]` and `queries[i]` consist only of only English letters.

Code Snippets

C++:

```
class Solution {
public:
vector<string> spellchecker(vector<string>& wordlist, vector<string>&
queries) {

}
};
```

Java:

```
class Solution {
public String[] spellchecker(String[] wordlist, String[] queries) {

}
}
```

Python3:

```
class Solution:
def spellchecker(self, wordlist: List[str], queries: List[str]) -> List[str]:
```