Problem 1772: Sort Features by Popularity

Problem Information

Difficulty: Medium
Acceptance Rate: 66.13%
Paid Only: Yes
Tags: Array, Hash Table, String, Sorting

Problem Description

You are given a string array `features` where `features[i]` is a single word that represents the name of a feature of the latest product you are working on. You have made a survey where users have reported which features they like. You are given a string array `responses`, where each `responses[i]` is a string containing space-separated words.

The **popularity** of a feature is the number of `responses[i]` that contain the feature. You want to sort the features in non-increasing order by their popularity. If two features have the same popularity, order them by their original index in `features`. Notice that one response could contain the same feature multiple times; this feature is only counted once in its popularity.

Return _the features in sorted order._

Example 1:

Input: features = ["cooler","lock","touch"], responses = ["i like cooler cooler","lock touch cool","locker like touch"] **Output:** ["touch","cooler","lock"] **Explanation:** appearances("cooler") = 1, appearances("lock") = 1, appearances("touch") = 2. Since "cooler" and "lock" both had 1 appearance, "cooler" comes first because "cooler" came first in the features array.

Example 2:

Input: features = ["a","aa","b","c"], responses = ["a","a aa","a a a a a","b a"] **Output:** ["a","aa","b","c"]

Constraints:

* `1 <= features.length <= 104` * `1 <= features[i].length <= 10` * `features` contains no duplicates. * `features[i]` consists of lowercase letters. * `1 <= responses.length <= 102` * `1 <= responses[i].length <= 103` * `responses[i]` consists of lowercase letters and spaces. * `responses[i]` contains no two consecutive spaces. * `responses[i]` has no leading or trailing spaces.

Code Snippets

C++:

```
class Solution {
public:
vector<string> sortFeatures(vector<string>& features, vector<string>&
responses) {

}
};
```

Java:

```
class Solution {
public String[] sortFeatures(String[] features, String[] responses) {

}
}
```

Python3:

```
class Solution:
def sortFeatures(self, features: List[str], responses: List[str]) ->
List[str]:
```