# Problem 916: Word Subsets

## Problem Information

**Difficulty:** Medium
**Acceptance Rate:** 55.85%
**Paid Only:** No
**Tags:** Array, Hash Table, String

## Problem Description

You are given two string arrays `words1` and `words2`.

A string `b` is a **subset** of string `a` if every letter in `b` occurs in `a` including multiplicity.

* For example, `"wrr"` is a subset of `"warrior"` but is not a subset of `"world"`.

A string `a` from `words1` is **universal** if for every string `b` in `words2`, `b` is a subset of `a`.

Return an array of all the **universal** strings in `words1`. You may return the answer in **any order**.

**Example 1:**

**Input:** words1 = ["amazon","apple","facebook","google","leetcode"], words2 = ["e","o"]

**Output:** ["facebook","google","leetcode"]

**Example 2:**

**Input:** words1 = ["amazon","apple","facebook","google","leetcode"], words2 = ["lc","eo"]

**Output:** ["leetcode"]

**Example 3:**

**Input:** words1 = ["acaac","cccbb","aacbb","caacc","bcbbb"], words2 = ["c","cc","b"]

**Output:** ["cccbb"]

**Constraints:**

* `1 <= words1.length, words2.length <= 104` * `1 <= words1[i].length, words2[i].length <= 10` * `words1[i]` and `words2[i]` consist only of lowercase English letters. * All the strings of `words1` are **unique**.

## Code Snippets

**C++:**

```
class Solution {
public:
vector<string> wordSubsets(vector<string>& words1, vector<string>& words2) {

}
};
```

**Java:**

```
class Solution {
public List<String> wordSubsets(String[] words1, String[] words2) {

}
}
```

**Python3:**

```
class Solution:
def wordSubsets(self, words1: List[str], words2: List[str]) -> List[str]:
```