Problem 1858: Longest Word With All Prefixes

Problem Information

Difficulty: Medium
Acceptance Rate: 71.91%
Paid Only: Yes
Tags: Array, String, Depth-First Search, Trie

Problem Description

Given an array of strings `words`, find the **longest** string in `words` such that **every prefix** of it is also in `words`.

* For example, let `words = ["a", "app", "ap"]`. The string `"app"` has prefixes `"ap"` and `"a"`, all of which are in `words`.

Return _the string described above. If there is more than one string with the same length, return the**lexicographically smallest** one, and if no string exists, return _`""`.

Example 1:

Input: words = ["k","ki","kir","kira", "kiran"] **Output:** "kiran" **Explanation:** "kiran" has prefixes "kira", "kir", "ki", and "k", and all of them appear in words.

Example 2:

Input: words = ["a", "banana", "app", "appl", "ap", "apply", "apple"] **Output:** "apple" **Explanation:** Both "apple" and "apply" have all their prefixes in words. However, "apple" is lexicographically smaller, so we return that.

Example 3:

Input: words = ["abc", "bc", "ab", "qwe"] **Output:** ""

Constraints:

* `1 <= words.length <= 105` * `1 <= words[i].length <= 105` * `1 <= sum(words[i].length) <= 105` * `words[i]` consists only of lowercase English letters.

Code Snippets

C++:

```
class Solution {
public:
string longestWord(vector<string>& words) {

}
};
```

Java:

```
class Solution {
public String longestWord(String[] words) {

}
}
```

Python3:

```
class Solution:
def longestWord(self, words: List[str]) -> str:
```