Problem 792: Number of Matching Subsequences

Problem Information

Difficulty: Medium
Acceptance Rate: 50.64%
Paid Only: No
Tags: Array, Hash Table, String, Binary Search, Dynamic Programming, Trie, Sorting

Problem Description

Given a string `s` and an array of strings `words`, return _the number of_ `words[i]` _that is a subsequence of_ `s`.

A **subsequence** of a string is a new string generated from the original string with some characters (can be none) deleted without changing the relative order of the remaining characters.

* For example, `"ace"` is a subsequence of `"abcde"`.

Example 1:

Input: s = "abcde", words = ["a","bb","acd","ace"] **Output:** 3 **Explanation:** There are three strings in words that are a subsequence of s: "a", "acd", "ace".

Example 2:

Input: s = "dsahjpjauf", words = ["ahjpjau","ja","ahbwzgqnuk","tnmlanowax"] **Output:** 2

Constraints:

* `1 <= s.length <= 5 * 104` * `1 <= words.length <= 5000` * `1 <= words[i].length <= 50` * `s` and `words[i]` consist of only lowercase English letters.

Code Snippets

C++:

```
class Solution {
public:
int numMatchingSubseq(string s, vector<string>& words) {

}
};
```

Java:

```
class Solution {
public int numMatchingSubseq(String s, String[] words) {

}
}
```

Python3:

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