

[Description](#)
[Solution](#)
[Discuss \(999+\)](#)
[Submissions](#)
i C++

792. Number of Matching Subsequences

Medium

4151

177

Add to List

Share

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 = "dsahjppjauf"`, `words = ["ahjppjau","ja","ahbwzgnuk","tnmlanowax"]`

Output: 2

Constraints:

- $1 \leq s.length \leq 5 * 10^4$
- $1 \leq words.length \leq 5000$
- $1 \leq words[i].length \leq 50$
- `s` and `words[i]` consist of only lowercase English letters.

Accepted 171,138

Submissions 332,813

Seen this question in a real interview before?

Yes

No

Companies



Related Topics

```

1  class Solution
2  public:
3      int numMatchingSubseq(string s, vector<string> words) {
4          int n = s.size();
5          int m = words.size();
6          for (int i = 0; i < m; i++) {
7              string word = words[i];
8              int index1 = 0, index2 = 0;
9              while (index1 < n && index2 < word.size()) {
10                 if (s[index1] == word[index2]) index2++;
11                 index1++;
12             }
13             if (index2 == word.size()) return ++count;
14         }
15         return count;
16     }
17 };
18
19
20
21
22
23
24
25
26
27
28

```

Your previous code was

Testcase

Run Code Re

Accepted

Runtime

Your input

"ab
["a

Output

3

Expected

3

Problems

Pick One

< Prev

792/2344

Next >

Example cases

?

Run C