Unknown Title

Description

Description

Note

Note

×

Editorial

Editorial

Solutions

Solutions

Submissions

Submissions

Code

Code

Testcase

Testcase

Test Result

Test Result

Leet

Leet

×

30. Substring with Concatenation of All Words

Hard

Topics
Companies

You are given a string `s` and an array of strings `words`. All the strings of `words` are of **the same length**.

A **concatenated string** is a string that exactly contains all the strings of any permutation of `words` concatenated.

- For example, if `words = ["ab","cd","ef"]`, then `"abcdef"`, `"abefcd"`, `"cdabef"`, `"cdefab"`, `"efabcd"`, and `"efcdab"` are all concatenated strings. `"acdbef"` is not a concatenated string because it is not the concatenation of any permutation of `words`.

Return an array of *the starting indices* of all the concatenated substrings in `s`. You can return the answer in **any order**.

Example 1:

Input: s = "barfoothefoobarman", words = ["foo","bar"]

Output: [0,9]

Explanation:

The substring starting at 0 is `"barfoo"`. It is the concatenation of `["bar","foo"]` which is a permutation of `words`.
The substring starting at 9 is `"foobar"`. It is the concatenation of `["foo","bar"]` which is a permutation of `words`.

Example 2:

Input: s = "wordgoodgoodgoodbestword", words = ["word","good","best","word"]

Output: []

Explanation:

There is no concatenated substring.

Example 3:

Input: s = "barfoofoobarthefoobarman", words = ["bar","foo","the"]

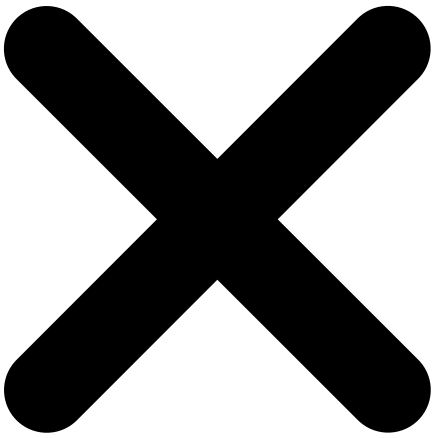
Output: [6,9,12]

Explanation:

The substring starting at 6 is `"foobarthe"`. It is the concatenation of `["foo","bar","the"]`.
The substring starting at 9 is `"barthefoo"`. It is the concatenation of `["bar","the","foo"]`.
The substring starting at 12 is `"thefoobar"`. It is the concatenation of `["the","foo","bar"]`.

Constraints:

- `1 <= s.length <= 10^4`
- `1 <= words.length <= 5000`
- `1 <= words[i].length <= 30`
- `s` and `words[i]` consist of lowercase English letters.



Seen this question in a real interview before?

1/5

Yes

No

Accepted

637,776/1.9M

Acceptance Rate

33.5%

Companies

Similar Questions

Minimum Window Substring

Hard

Discussion (270)

</>
@

💡 Discussion Rules

1. Please don't post **any solutions** in this discussion.

2. The problem discussion is for asking questions about the problem or for sharing tips - anything except for solutions.

3. If you'd like to share your solution for feedback and ideas, please head to the solutions tab and post it there.

Daman

Feb 28, 2024

Failing on test case
"aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa"

Me:
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

Read more

356

Show 12 Replies

Reply

Abhishek kumar

Sep 13, 2021

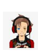
Description of question is not good , various points are missing

Read more

194

Show 5 Replies

Reply



Abdulaziz

Nov 29, 2023

178 / 179 testcases passed 🤣

Read more

97

Show 4 Replies

Reply

cornellouis

Jun 06, 2018

I thought the question assumed all words were the same length?

Read more

138

Show 7 Replies

Reply

zwright8

Dec 14, 2022

Can LeetCode start adding the efficiency requirements to all the Description sections? It would make problem solving much less tedious

Read more

94

Show 4 Replies

Reply

Dany__Guty

Feb 12, 2023

This will save you: you need to increment the window one by one, there's no way to pass the testcases in increments in the window of n = words[0].size()

Read more

56

Show 3 Replies

Reply

leolin479

Dec 23, 2021

For the example case

Input: s = "wordgoodgoodgoodbestword", words = ["word","good","best","word"]
Output: []

The standard answer was "output: [8]", which preventing my code to be submitted.

Read more

41

Show 7 Replies

Reply

darcylmx

Dec 30, 2019

Ignoring the key information(all the words are the SAME length) will cause exponential time complexity.

Otherwise, it is just linear time to check a match.

Read more

34

Show 3 Replies

Reply

Igor

Nov 27, 2023

Additional description point:
The words can overlap, like in this example from the test cases:
s =
"lingmindraboofooowingdingbarrwingmonkeypoundcake"
words =
["fooo","barr","wing","ding","wing"]

Please, add this to the description, it is not clear

Read more

50

Show 2 Replies

Reply

Yash-Dev-Solanki

Aug 16, 2023

There is a problem with the last testcase. The parameters seem to be missing.

Read more

23

Reply

113 Online

1

2

3

4

5

```
class Solution {
    public List<Integer> findSubstring(String s, String[] words) {


    }
}
```

Saved

Ln 1, Col 1

s =

"barfoothefoobarman"

words =

["foo","bar"]

9

1

2

3

4

5

6

›

"barfoothefoobarman"

["foo","bar"]

"wordgoodgoodgoodbestword"

["word","good","best","word"]

"barfoofoobarthefoobarman"

["bar","foo","the"]
</>
Source

😎

Stuck? Leet guides you through every line.

Subscribe to Premium

Qwen
Qwen-Flash

FindHeaderBarSize

FindTabBarSize

FindBorderBarSize