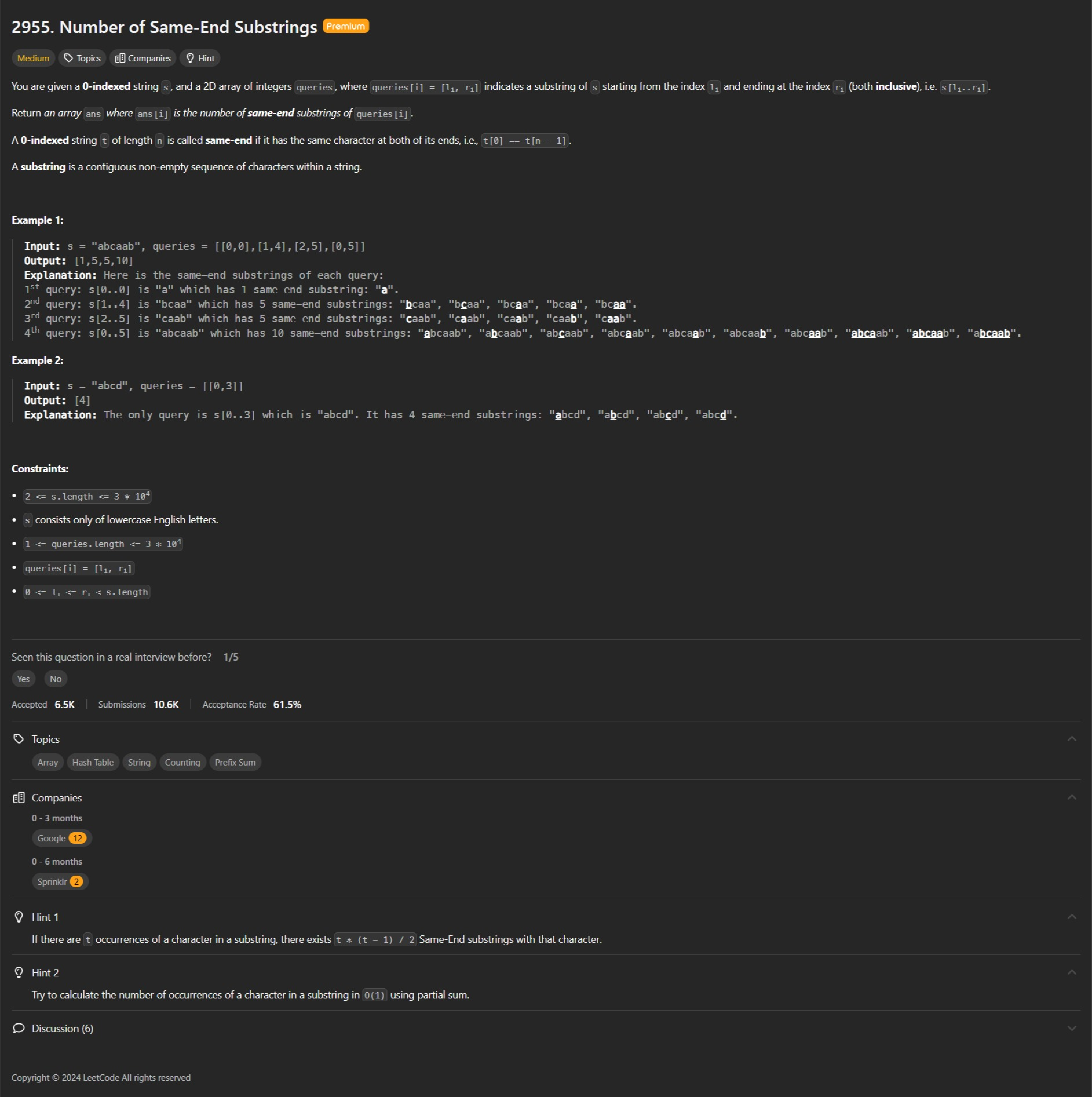

2955. Number of Same-End Substrings

You are given a **0-indexed** string `s`, and a 2D array of integers `queries`, where `queries[i]` = [l_i, r_i] indicates a substring of `s` starting from the index l_i and ending at the index r_i (both **inclusive**), i.e. `s`[l_i..r_i].

Return *an array* `ans` *where* `ans[i]` *is the number of* ***same-end*** *substrings of* `queries[i]`.

A **0-indexed** string `t` of length `n` is called **same-end** if it has the same character at both of its ends, i.e., `t[0] == t[n - 1]`.

A **substring** is a contiguous non-empty sequence of characters within a string.

Example 1:

```
Input: s = "abcaab", queries = [[0,0],[1,4],[2,5],[0,5]]
Output: [1,5,5,10]
Explanation: Here is the same-end substrings of each query:
1st query: s[0..0] is "a" which has 1 same-end substring: "a".
2nd query: s[1..4] is "bcaa" which has 5 same-end substrings: "bcaa", "bcaa", "bcaa", "bcaa", "bcaa".
3rd query: s[2..5] is "caab" which has 5 same-end substrings: "caab", "caab", "caab", "caab", "caab".
4th query: s[0..5] is "abcaab" which has 10 same-end substrings: "abcaab", "abcaab", "abcaab", "abcaab", "abcaab", "abcaab", "abcaab", "abcaab", "abcaab", "abcaab".
```

Example 2:

```
Input: s = "abcd", queries = [[0,3]]
Output: [4]
Explanation: The only query is s[0..3] which is "abcd". It has 4 same-end substrings: "abcd", "abcd", "abcd", "abcd".
```

Constraints:

- $2 <= s.length <= 3 * 10^4$
- `s` consists only of lowercase English letters.
- $1 <= queries.length <= 3 * 10^4$
- `queries[i]` = [l_i, r_i]
- $0 <= l_i <= r_i < s.length$

Seen this question in a real interview before? 1/5

Accepted 6.5K | Submissions 10.6K | Acceptance Rate 61.5%

Topics

Companies

0 - 3 months

Google 12

0 - 6 months

Sprinklr 2

Hint 1

If there are `t` occurrences of a character in a substring, there exists `t * (t - 1) / 2` Same-End substrings with that character.

Hint 2

Try to calculate the number of occurrences of a character in a substring in `O(1)` using partial sum.

Discussion (6)