

1698. Number of Distinct Substrings in a String

Medium👍 75🔒 18❤️ Add to List🔖 Share

Given a string `s`, return *the number of **distinct** substrings* of `s`.

A **substring** of a string is obtained by deleting any number of characters (possibly zero) from the front of the string and any number (possibly zero) from the back of the string.

Example 1:

Input: `s = "aabbaba"`
Output: 21
Explanation: The set of distinct strings is
["a","b","aa","bb","ab","ba","aab","abb","bab","bba","aba","aabb","abba","bbab","bab"]

Example 2:

Input: `s = "abcdefg"`
Output: 28

Constraints:

- `1 <= s.length <= 500`
- `s` consists of lowercase English letters.

Follow up: Can you solve this problem in $O(n)$ time complexity?

Accepted 3,196 | Submissions 5,144

Seen this question in a real interview before?

Companies 🏢 *i* ^

0 ~ 6 months 6 months ~ 1 year 1 year ~ 2 years

Related Topics ^

[String](#) [Trie](#) [Rolling Hash](#) [Suffix Array](#) [Hash Function](#)

Hide Hint 1 ^

Calculate the prefix hashing array for `s`.

Hide Hint 2 ^

Use the prefix hashing array to calculate the hashing value of each substring.

Hide Hint 3 ^

Compare the hashing values to determine the unique substrings.

Hide Hint 4 ^

There could be collisions if you use hashing, what about double hashing.

```
1 class Solution {
2     public int countDistinct(String s) {
3
4     }
5 }
```

