Problem 1698: Number of Distinct Substrings in a String

Problem Information

Difficulty: Medium
Acceptance Rate: 64.63%
Paid Only: Yes
Tags: String, Trie, Rolling Hash, Suffix Array, Hash Function

Problem Description

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","baba","aabba","abbab","bbaba","aabbab","abbaba","aabbaba"]

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?

Code Snippets

C++:

```
class Solution {
public:
int countDistinct(string s) {

}
};
```

Java:

```
class Solution {
public int countDistinct(String s) {

}
}
```

Python3:

```
class Solution:
def countDistinct(self, s: str) -> int:
```