Problem 2223: Sum of Scores of Built Strings

Problem Information

Difficulty: Hard
Acceptance Rate: 43.63%
Paid Only: No
Tags: String, Binary Search, Rolling Hash, Suffix Array, String Matching, Hash Function

Problem Description

You are **building** a string `s` of length `n` **one** character at a time, **prepending** each new character to the **front** of the string. The strings are labeled from `1` to `n`, where the string with length `i` is labeled `si`.

* For example, for `s = "abaca"`, `s1 == "a"`, `s2 == "ca"`, `s3 == "aca"`, etc.

The **score** of `si` is the length of the **longest common prefix** between `si` and `sn` (Note that `s == sn`).

Given the final string `s`, return _the**sum** of the **score** of every _`si`.

Example 1:

Input: s = "babab" **Output:** 9 **Explanation:** For s1 == "b", the longest common prefix is "b" which has a score of 1. For s2 == "ab", there is no common prefix so the score is 0. For s3 == "bab", the longest common prefix is "bab" which has a score of 3. For s4 == "abab", there is no common prefix so the score is 0. For s5 == "babab", the longest common prefix is "babab" which has a score of 5. The sum of the scores is 1 + 0 + 3 + 0 + 5 = 9, so we return 9.

Example 2:

Input: s = "azbazbzaz" **Output:** 14 **Explanation:** For s2 == "az", the longest common prefix is "az" which has a score of 2. For s6 == "azbzaz", the longest common prefix is "azb" which has a score of 3. For s9 == "azbazbzaz", the longest common prefix is "azbazbzaz" which has a score of 9. For all other si, the score is 0. The sum of the scores is 2 + 3 + 9 = 14, so we return 14.

Constraints:

* `1 <= s.length <= 105` * `s` consists of lowercase English letters.

Code Snippets

C++:

```
class Solution {
public:
long long sumScores(string s) {

}
};
```

Java:

```
class Solution {
public long sumScores(String s) {

}
}
```

Python3:

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