Problem 467: Unique Substrings in Wraparound String

Problem Information

Difficulty: Medium
Acceptance Rate: 42.22%
Paid Only: No
Tags: String, Dynamic Programming

Problem Description

We define the string `base` to be the infinite wraparound string of `"abcdefghijklmnopqrstuvwxyz"`, so `base` will look like this:

* `"...zabcdefghijklmnopqrstuvwxyzabcdefghijklmnopqrstuvwxyzabcd...."`.

Given a string `s`, return _the number of**unique non-empty substrings** of _`s` _are present in_`base`.

Example 1:

Input: s = "a" **Output:** 1 **Explanation:** Only the substring "a" of s is in base.

Example 2:

Input: s = "cac" **Output:** 2 **Explanation:** There are two substrings ("a", "c") of s in base.

Example 3:

Input: s = "zab" **Output:** 6 **Explanation:** There are six substrings ("z", "a", "b", "za", "ab", and "zab") of s in base.

Constraints:

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

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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