Problem 3031: Minimum Time to Revert Word to Initial State II

Problem Information

Difficulty: Hard
Acceptance Rate: 34.80%
Paid Only: No
Tags: String, Rolling Hash, String Matching, Hash Function

Problem Description

You are given a **0-indexed** string `word` and an integer `k`.

At every second, you must perform the following operations:

* Remove the first `k` characters of `word`. * Add any `k` characters to the end of `word`.

Note that you do not necessarily need to add the same characters that you removed. However, you must perform **both** operations at every second.

Return _the**minimum** time greater than zero required for_ `word` _to revert to its**initial** state_.

Example 1:

Input: word = "abacaba", k = 3 **Output:** 2 **Explanation:** At the 1st second, we remove characters "aba" from the prefix of word, and add characters "bac" to the end of word. Thus, word becomes equal to "cababac". At the 2nd second, we remove characters "cab" from the prefix of word, and add "aba" to the end of word. Thus, word becomes equal to "abacaba" and reverts to its initial state. It can be shown that 2 seconds is the minimum time greater than zero required for word to revert to its initial state.

Example 2:

Input: word = "abacaba", k = 4 **Output:** 1 **Explanation:** At the 1st second, we remove characters "abac" from the prefix of word, and add characters "caba" to the end of

word. Thus, word becomes equal to "abacaba" and reverts to its initial state. It can be shown that 1 second is the minimum time greater than zero required for word to revert to its initial state.

Example 3:

Input: word = "abcbabcd", k = 2 **Output:** 4 **Explanation:** At every second, we will remove the first 2 characters of word, and add the same characters to the end of word. After 4 seconds, word becomes equal to "abcbabcd" and reverts to its initial state. It can be shown that 4 seconds is the minimum time greater than zero required for word to revert to its initial state.

Constraints:

* `1 <= word.length <= 106` * `1 <= k <= word.length` * `word` consists only of lowercase English letters.

Code Snippets

C++:

```cpp
class Solution {
public:
int minimumTimeToInitialState(string word, int k) {


}
};
```

Java:

```java
class Solution {
public int minimumTimeToInitialState(String word, int k) {


}
}
```

Python3:

```python
class Solution:
def minimumTimeToInitialState(self, word: str, k: int) -> int:
```