Problem 2379: Minimum Recolors to Get K Consecutive Black Blocks

Problem Information

Difficulty: Easy
Acceptance Rate: 68.47%
Paid Only: No
Tags: String, Sliding Window

Problem Description

You are given a **0-indexed** string `blocks` of length `n`, where `blocks[i]` is either `'W'` or `'B'`, representing the color of the `ith` block. The characters `'W'` and `'B'` denote the colors white and black, respectively.

You are also given an integer `k`, which is the desired number of **consecutive** black blocks.

In one operation, you can **recolor** a white block such that it becomes a black block.

Return _the**minimum** number of operations needed such that there is at least **one** occurrence of _`k` _consecutive black blocks._

Example 1:

Input: blocks = "WBBWWBBWBW", k = 7 **Output:** 3 **Explanation:** One way to achieve 7 consecutive black blocks is to recolor the 0th, 3rd, and 4th blocks so that blocks = "BBBBBBBWBW". It can be shown that there is no way to achieve 7 consecutive black blocks in less than 3 operations. Therefore, we return 3.

Example 2:

Input: blocks = "WBWBBBW", k = 2 **Output:** 0 **Explanation:** No changes need to be made, since 2 consecutive black blocks already exist. Therefore, we return 0.

Constraints:

* `n == blocks.length` * `1 <= n <= 100` * `blocks[i]` is either `'W'` or `'B'`. * `1 <= k <= n`

Code Snippets

C++:

```
class Solution {
public:
int minimumRecolors(string blocks, int k) {

}
};
```

Java:

```
class Solution {
public int minimumRecolors(String blocks, int k) {

}
}
```

Python3:

```
class Solution:
def minimumRecolors(self, blocks: str, k: int) -> int:
```