Problem 1151: Minimum Swaps to Group All 1's Together

Problem Information

Difficulty: Medium
Acceptance Rate: 61.15%
Paid Only: Yes
Tags: Array, Sliding Window

Problem Description

Given a binary array `data`, return the minimum number of swaps required to group all `1`'s present in the array together in **any place** in the array.

Example 1:

Input: data = [1,0,1,0,1] **Output:** 1 **Explanation:** There are 3 ways to group all 1's together: [1,1,1,0,0] using 1 swap. [0,1,1,1,0] using 2 swaps. [0,0,1,1,1] using 1 swap. The minimum is 1.

Example 2:

Input: data = [0,0,0,1,0] **Output:** 0 **Explanation:** Since there is only one 1 in the array, no swaps are needed.

Example 3:

Input: data = [1,0,1,0,1,0,0,1,1,0,1] **Output:** 3 **Explanation:** One possible solution that uses 3 swaps is [0,0,0,0,0,1,1,1,1,1,1].

Constraints:

* `1 <= data.length <= 105` * `data[i]` is either `0` or `1`.

Code Snippets

C++:

```
class Solution {
public:
int minSwaps(vector<int>& data) {

}
};
```

Java:

```
class Solution {
public int minSwaps(int[] data) {

}
}
```

Python3:

```
class Solution:
def minSwaps(self, data: List[int]) -> int:
```