Description | Solution | Discuss (273) | Submissions

1151. Minimum Swaps to Group All 1's Together

Medium 642 7 Add to List Share

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 <= 10^5`
- `data[i]` is either `0` or `1`.

Accepted 27,668 Submissions 46,469

C#

```
public class Solution {
    public int MinSwaps(int[] data) {
        var windowSize = data.Count(x => x == 1);

        if(windowSize == 0)
        {
            return 0;
        }

        var currOnes = 0;
        var maxOnes = 0;
        for(int i =0; i < data.Length; i++)
        {
            currOnes += data[i];
            if(i >= windowSize)
            {
                currOnes -= data[i- windowSize];
            }
            maxOnes = Math.Max(maxOnes, currOnes);
        }

       return windowSize == 1 ? 0 : maxOnes;

    }
}
```

Testcase | Run Code Result

Accepted Runtime: 84 ms

Your input `[1,0,1,0,1]`

Output `1`

Expected `1`

Console | Use Example Testcase