

Separate Black And White Balls

🕒 solved by	Senan
🌐 Platform	LeetCode
🔧 difficulty	Medium
# Serial	2938
≡ tags	Logic Maths String Manipulation
🗨 language	C++
📅 solved on	@15/10/2024
🔗 link	https://leetcode.com/problems/separate-black-and-white-balls/description/
☑ Completion	✓

Intuition

We are given a binary string `s` representing black and white balls. Our goal is to group all black balls (represented by '1') on the right and all white balls (represented by '0') on the left. The key observation is that to minimize the number of swaps, each black ball ('1') should be moved past all white balls ('0') that lie to its right. So for each black ball, we count how many white balls are to its right and calculate how many swaps are needed.

Approach

- We traverse the string from right to left.
- Maintain a counter `zeros` to track how many white balls ('0') we have seen so far as we iterate.
- For each black ball ('1') we encounter, add the current `zeros` value to the result because this represents the number of steps required to move the black ball past all the white balls to its right.
- This ensures we minimize the number of swaps by always moving the black balls past white balls with minimal steps.

Complexity

Time Complexity:

- We iterate through the string once, so the time complexity is $O(n)$, where `n` is the length of the string.

Space Complexity:

- We use only a few variables (`answer` and `zeros`), so the space complexity is $O(1)$, constant space.

Code

```
class Solution {
public:
    long long minimumSteps(string s) {
```

```
long long answer = 0;
int zeros = 0;
for (int i = s.size() - 1; i >= 0; i--) {
    if (s[i] == '0') zeros++;
    else answer += zeros;
}
return answer;
}
};
```