Problem 1963: Minimum Number of Swaps to Make the String Balanced

Problem Information

Difficulty: Medium
Acceptance Rate: 78.08%
Paid Only: No
Tags: Two Pointers, String, Stack, Greedy

Problem Description

You are given a **0-indexed** string `s` of **even** length `n`. The string consists of **exactly** `n / 2` opening brackets `'['` and `n / 2` closing brackets `']'`.

A string is called **balanced** if and only if:

* It is the empty string, or * It can be written as `AB`, where both `A` and `B` are **balanced** strings, or * It can be written as `[C]`, where `C` is a **balanced** string.

You may swap the brackets at **any** two indices **any** number of times.

Return _the**minimum** number of swaps to make _`s` _**balanced**_.

Example 1:

Input: s = "][][" **Output:** 1 **Explanation:** You can make the string balanced by swapping index 0 with index 3. The resulting string is "[[]]".

Example 2:

Input: s = "]]][[[" **Output:** 2 **Explanation:** You can do the following to make the string balanced: - Swap index 0 with index 4. s = "[]][[". - Swap index 1 with index 5. s = "[[][]]". The resulting string is "[[][]]".

Example 3:

Input: s = "[]" **Output:** 0 **Explanation:** The string is already balanced.

Constraints:

* `n == s.length` * `2 <= n <= 106` * `n` is even. * `s[i]` is either `'[' `or `']'`. * The number of opening brackets `'['` equals `n / 2`, and the number of closing brackets `']'` equals `n / 2`.

Code Snippets

C++:

```
class Solution {
public:
int minSwaps(string s) {

}
};
```

Java:

```
class Solution {
public int minSwaps(String s) {

}
}
```

Python3:

```
class Solution:
def minSwaps(self, s: str) -> int:
```