Problem 1529: Minimum Suffix Flips

Problem Information

Difficulty: Medium
Acceptance Rate: 73.79%
Paid Only: No
Tags: String, Greedy

Problem Description

You are given a **0-indexed** binary string `target` of length `n`. You have another binary string `s` of length `n` that is initially set to all zeros. You want to make `s` equal to `target`.

In one operation, you can pick an index `i` where `0 <= i < n` and flip all bits in the **inclusive** range `[i, n - 1]`. Flip means changing `'0'` to `'1'` and `'1'` to `'0'`.

Return _the minimum number of operations needed to make_`s` _equal to_`target`.

Example 1:

Input: target = "10111" **Output:** 3 **Explanation:** Initially, s = "00000". Choose index i = 2: "00 _000_ " -> "00 _111_ " Choose index i = 0: "_00111_ " -> "_11000_ " Choose index i = 1: "1 _1000_ " -> "1 _0111_ " We need at least 3 flip operations to form target.

Example 2:

Input: target = "101" **Output:** 3 **Explanation:** Initially, s = "000". Choose index i = 0: "_000_ " -> "_111_ " Choose index i = 1: "1 _11_ " -> "1 _00_ " Choose index i = 2: "10 _0_ " -> "10 _1_ " We need at least 3 flip operations to form target.

Example 3:

Input: target = "00000" **Output:** 0 **Explanation:** We do not need any operations since the initial s already equals target.

Constraints:

* `n == target.length` * `1 <= n <= 105` * `target[i]` is either `'0'` or `'1'`.

Code Snippets

C++:

```
class Solution {
public:
int minFlips(string target) {

}
};
```

Java:

```
class Solution {
public int minFlips(String target) {

}
}
```

Python3:

```
class Solution:
def minFlips(self, target: str) -> int:
```