6298. Apply Bitwise Operations to Make Strings Equal

My Submissions (/contest/weekly-contest-329/problems/apply-bitwise-operations-to-make-strings-equal/submissions/)

Back to Contest (/contest/weekly-contest-329/)

You are given two **0-indexed binary** strings `s` and `target` of the same length `n`. You can do the following operation on `s` **any** number of times:

- Choose two **different** indices `i` and `j` where `0 <= i, j < n`.
- Simultaneously, replace `s[i]` with (`s[i]` **OR** `s[j]`) and `s[j]` with (`s[i]` **XOR** `s[j]`).

For example, if `s = "0110"`, you can choose `i = 0` and `j = 2`, then simultaneously replace `s[0]` with (`s[0]` **OR** `s[2] = 0` **OR** `1 = 1`), and `s[2]` with (`s[0]` **XOR** `s[2] = 0` **XOR** `1 = 1`), so we will have `s = "1110"`.

Return `true` *if you can make the string* `s` *equal to* `target`, *or* `false` *otherwise.*

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Medium

Example 1:

```
Input: s = "1010", target = "0110"
Output: true
Explanation: We can do the following operations:
- Choose i = 2 and j = 0. We have now s = "0010".
- Choose i = 2 and j = 1. We have now s = "0110".
Since we can make s equal to target, we return true.
```

Example 2:

```
Input: s = "11", target = "00"
Output: false
Explanation: It is not possible to make s equal to target with any number of operations.
```

Constraints:

- `n == s.length == target.length`
- `2 <= n <= 10^5`
- `s` and `target` consist of only the digits `0` and `1`.

JavaScript

```
const makeStringsEqual = (s, t) => {
    let flag1 = 0, flag2 = 0, n = s.length;
    for (let i = 0; i < n; i++) {
        if (s[i] == '1') flag1 = 1;
        if (t[i] == '1') flag2 = 1;
        if (flag1 && flag2) return true;
    }
    if (!flag1 && !flag2) return true;
    return false;
};
```

Use Example Testcases

Custom Testcase

Run Submit

Submission Result: Accepted (/submissions/detail/882817553/) More Details (/submissions/detail/882817553/)

Share your acceptance!

1