

844. Backspace String Compare

Given two strings *s* and *t*, return true *if they are equal when both are typed into empty text editors*. '#' means a backspace character.

Note that after backspacing an empty text, the text will continue empty.

Example 1:

Input: *s* = "ab#c", *t* = "ad#c"

Output: true

Explanation: Both *s* and *t* become "ac".

Example 2:

Input: *s* = "ab##", *t* = "c#d#"

Output: true

Explanation: Both *s* and *t* become "".

Example 3:

Input: *s* = "a#c", *t* = "b"

Output: false

Explanation: *s* becomes "c" while *t* becomes "b".

Constraints:

- $1 \leq s.length, t.length \leq 200$
- *s* and *t* only contain lowercase letters and '#' characters.

```
class Solution {
public:
    string processString(const string& s){
        string result;
        for(char ch : s){
            if(ch == '#'){
                if(!result.empty())
                    result.pop_back();
            }
            else{
                result.push_back(ch);
            }
        }
        return result;
    }
    bool backspaceCompare(string s, string t) {
        string finalS = processString(s);
        string finalT = processString(t);

        return finalS == finalT;
    }
};
```