**844. Backspace String Compare: -**

Easy Accepted: 701.3K Submissions: 1.4M Acceptance Rate: 48.4%

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 <= s.length, t.length <= 200
* s and t only contain lowercase letters and '#' characters.

**Follow up:** Can you solve it in O(n) time and O(1) space?

**Code: -**

class Solution {

public:

    bool backspaceCompare(string s, string t) {

        unordered\_map<string,bool> mp;

        string str = "";

        for(int i = 0; i < s.size(); ++i){

            if(s[i] != '#')     str.push\_back(s[i]);

            else if(s[i] == '#' and str.size())       str.pop\_back();

        }

        mp[str] = true;

        str = "";

        for(int i = 0; i < t.size(); ++i){

            if(t[i] != '#')     str.push\_back(t[i]);

            else if(t[i] == '#' and str.size())       str.pop\_back();

        }

        return mp[str] == true;

    }

};

**T.C: - O(max(N1, N2))**

**S.C: - O(N)**

**N1, N2 = length of s, t**