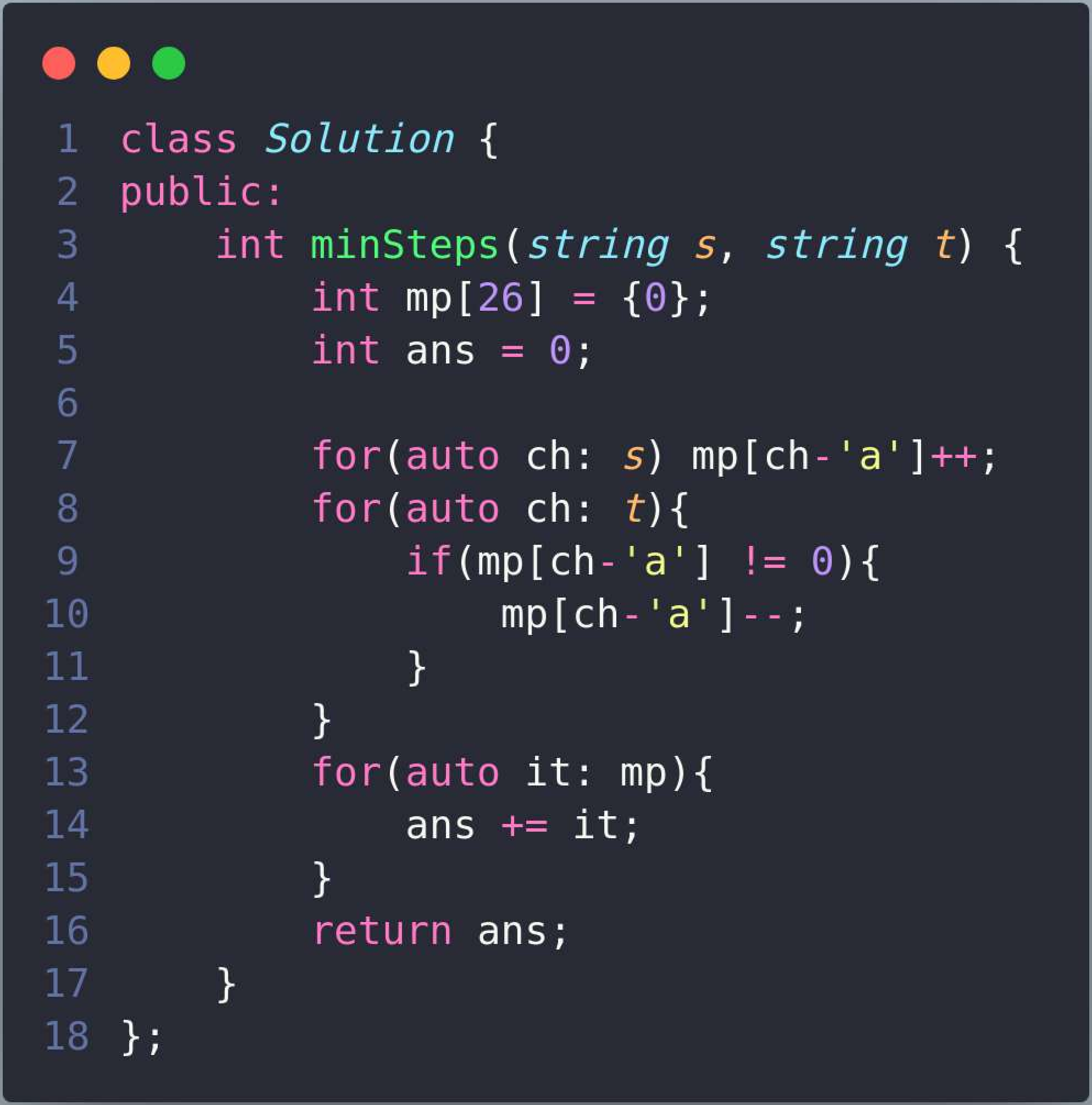
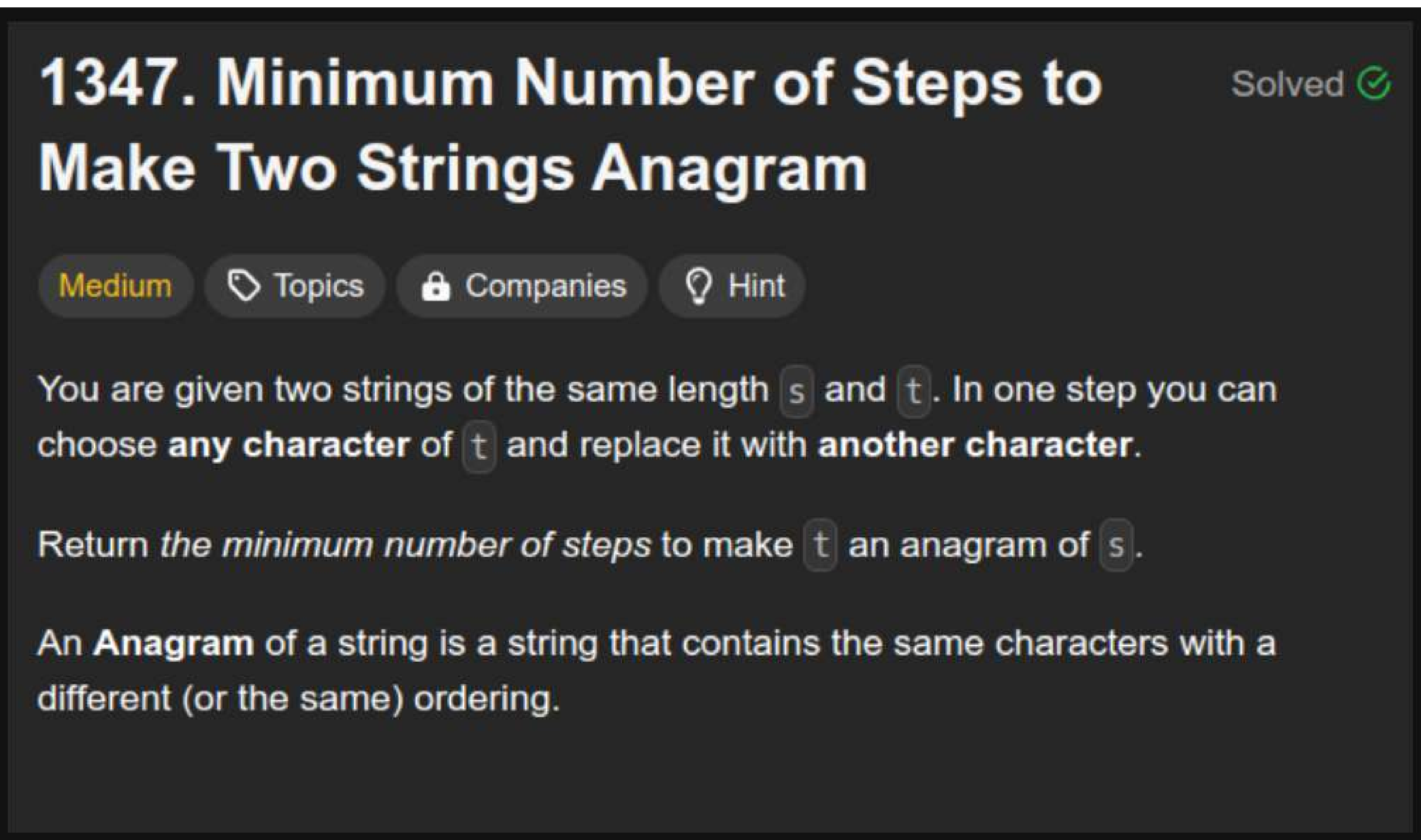


```
class Solution {
public:
    int minSteps(string s, string t) {
        int mp[26] = {0};
        int ans = 0;

        for(auto ch: s) mp[ch-'a']++;
        for(auto ch: t){
            if(mp[ch-'a'] != 0){
                mp[ch-'a']--;
            }
        }
        for(auto it: mp){
            ans += it;
        }
        return ans;
    }
};
```



1347. Minimum Number of Steps to Make Two Strings Anagram

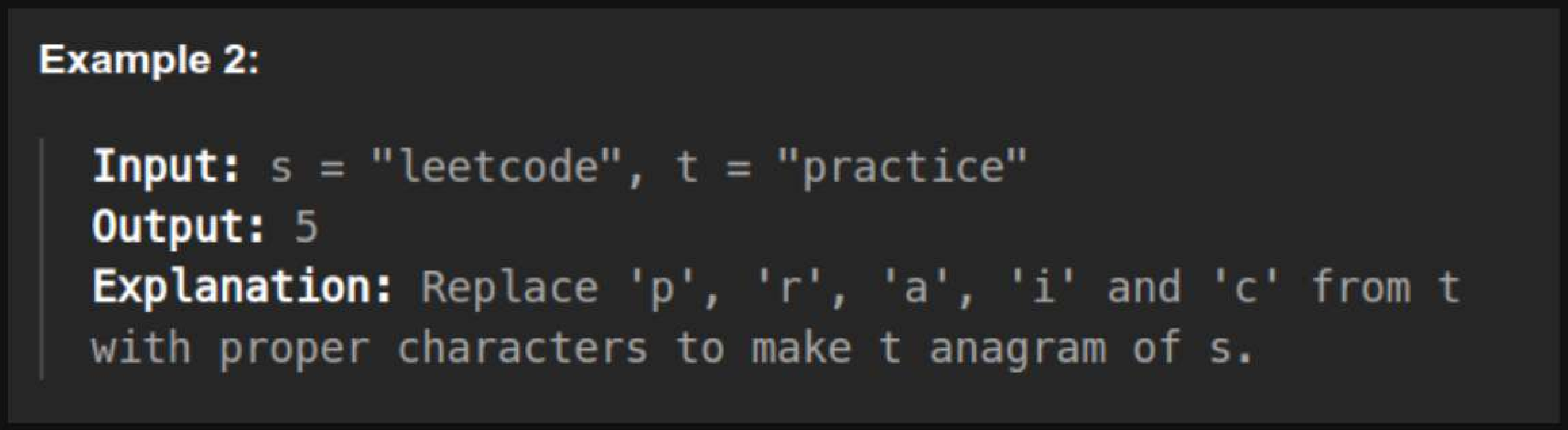
Solved

Medium | Topics | Companies | Hint

You are given two strings of the same length `s` and `t`. In one step you can choose **any character** of `t` and replace it with **another character**.

Return *the minimum number of steps* to make `t` an anagram of `s`.

An **Anagram** of a string is a string that contains the same characters with a different (or the same) ordering.

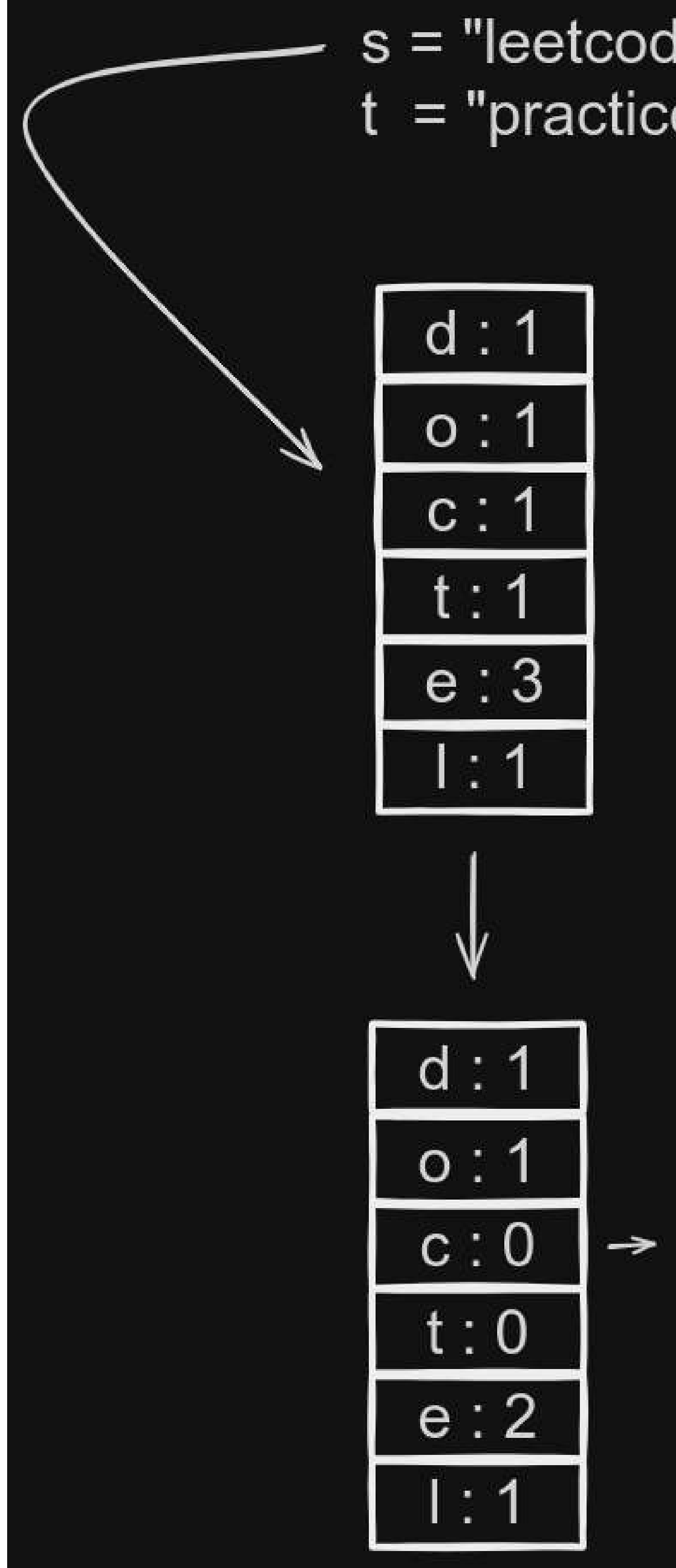


Example 2:

```
Input: s = "leetcode", t = "practice"
Output: 5
Explanation: Replace 'p', 'r', 'a', 'i' and 'c' from t
with proper characters to make t anagram of s.
```

Intuition
As we are sure there will be only lowercase characters so we will use 26 sized array to store frequency of characters.

Approach
1-> Create array of size 26 and fill with 0
2-> Iterate over s and store its character freq in array
3-> Iterate over t and decrement only those characters that are occuring in bith s and t
4-> NOTE: Once occurence is set to 0 we will not decrement it because it's useless
5-> Final answer will be sum of all remaining occurences in freq array



s = "leetcode"
t = "practice"

d : 1
o : 1
c : 1
t : 1
e : 3
l : 1

Observation:

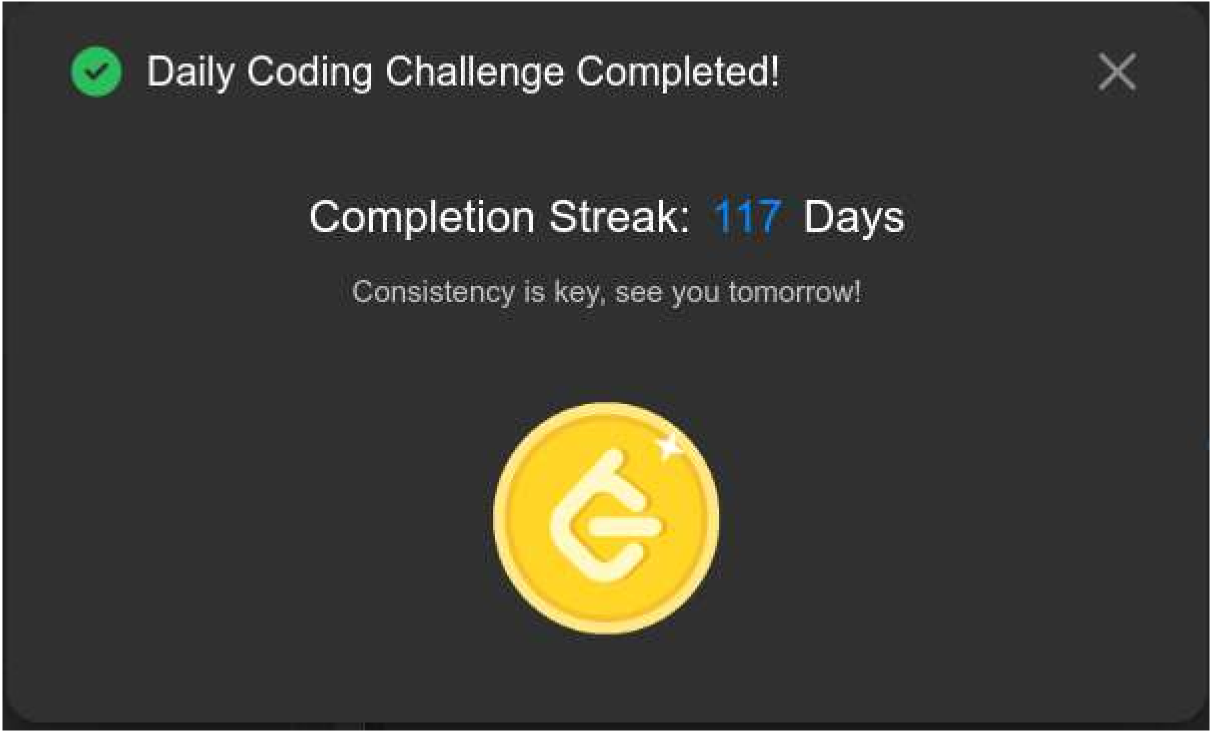
When we have stored occurences of s in our freq array we just need to decrease the count of only those characters that are present in s and also occuring in t string.

t = "practice"

d : 1
o : 1
c : 0
t : 0
e : 2
l : 1

c is occuring in t so decrement its count. One thing to notice is that once occurence is set to 0 we will not decrement it because it is useless and will affect our ans if we decrement 0.

ans = remaining occurences = 1 + 1 + 2 + 1 = 5



✓ Daily Coding Challenge Completed!

Completion Streak: 117 Days

Consistency is key, see you tomorrow!