

[<< Prev](#)[Next >>](#)[Problem](#)[Editorial](#)[Submissions](#)

## Anagram

Difficulty: **Easy**Accuracy: **44.93%**Submissions: **368K+**Points: **2**

Given two strings **s1** and **s2** consisting of lowercase characters. The task is to check whether two given strings are an anagram of each other or not. An anagram of a string is another string that contains the same characters, only the order of characters can be different. For example, "act" and "tac" are an anagram of each other. Strings **s1** and **s2** can only contain lowercase alphabets.

Note: You can assume both the strings s1 & s2 are **non-empty**.

### Examples :

**Input:** s1 = "geeks", s2 = "kseeg"

**Output:** true

**Explanation:** Both the string have same characters with same frequency. So, they are anagrams.

**Input:** s1 = "allergy", s2 = "allergic"

**Output:** false

**Explanation:** Characters in both the strings are not same, so they are not anagrams.

Java (1.8)

Average Time: 20m

[Start Timer](#)

```
1 // } Driver Code Ends
29
30
31 class Solution {
32     public static boolean areAnagrams(String s1, String s2) {
33         if (s1.length() != s2.length()) {
34             return false;
35         }
36         int[] charCount = new int[26];
37         for (int i = 0; i < s1.length(); i++) {
38             charCount[s1.charAt(i) - 'a']++;
39             charCount[s2.charAt(i) - 'a']--;
40         }
41         for (int count : charCount) {
42             if (count != 0) {
43                 return false;
44             }
45         }
46         return true;
47     }
48 }
49
```

[Custom Input](#)[Compile & Run](#)[Submit](#)