Valid Anagram

⊙ Difficulty	Easy	
⊙ Category	Arrays	
🔗 Question	https://leetcode.com/problems/valid-anagram/	
🔗 Solution	https://youtu.be/9UtInBqnCgA	
⋰ Status	Done	

Question

> Given two strings `s` and `t`, return `true` *if* `t` *is an anagram of* `s`, *and* `false` *otherwise*.
>
> An **Anagram** is a word or phrase formed by rearranging the letters of a different word or phrase, typically using all the original letters exactly once.

Example

Example 1:

```
Input: s = "anagram", t = "nagaram"
Output: true
```

Example 2:

```
Input: s = "rat", t = "car"
Output: false
```

Idea

💡 If count frequency of two strings are the same, true; otherwise false → O(n log n)

💡 HashMap→ O(n)

Solution

```python
class Solution:
    def isAnagram(self, s: str, t: str) -> bool:
        sorted_s = sorted(s)
        sorted_t = sorted(t)
        return sorted_s == sorted_t
```

```python
class Solution:
    def isAnagram(self, s: str, t: str) -> bool:
        count = defaultdict(int)

        # Count the frequency of characters in string s
        for x in s:
            count[x] += 1

        # Decrement the frequency of characters in string t
        for x in t:
            count[x] -= 1

        # Check if any character has non-zero frequency
        for val in count.values():
            if val != 0:
                return False

        return True
```