Valid Anagram (Easy)

Given two strings `s` and `t`, return `true` if the two strings are anagrams of each other, otherwise return `false`.

An anagram is a string that contains the exact same characters as another string, but the order of the characters can be different.

Example 1:

```
Input: s = "racecar", t = "carrace"

Output: true
```

Example 2:

```
Input: s = "jar", t = "jam"

Output: false
```

Constraints:

`s` and `t` consist of lowercase English letters.

思路:

1.排序後比對T:O(NlogN), S:O(1)→最多26個字母
2.將出現字母存儲出現次數比對 O(n), S:O(1)

作法一

```
class Solution {
public:
    bool isAnagram(string s, string t) {
        if(s.length()!=t.length()){
            return false;
        }
        sort(s.begin(),s.end());
```

```
        sort(t.begin(),t.end());
        return s==t;
    }
};
```

作法2

```
class Solution {
public:
    bool isAnagram(string s, string t) {
        if(s.length()!=t.length()){
            return false;
        }
        int c[26]={0};
        for(int i=0;i<s.length();i++){
            c[s[i]-'a']++;
            c[t[i]-'a']--;
        }
        for(int val:c){
            if(val!=0){
                return false;
            }
        }
        return true;
    }
};
```