

```

1  /*
2  给定两个字符串 s 和 t ，编写一个函数来判断 t 是否是 s 的字母异位词。
3  示例 1:
4  输入: s = "anagram", t = "nagaram"
5  输出: true
6  示例 2:
7  输入: s = "rat", t = "car"
8  输出: false
9  说明:
10 你可以假设字符串只包含小写字母。
11 进阶:
12 如果输入字符串包含 unicode 字符怎么办? 你能否调整你的解法来应对这种情况?
13 来源: 力扣 (LeetCode)
14 链接: https://leetcode-cn.com/problems/valid-anagram
15 著作权归领扣网络所有。商业转载请联系官方授权, 非商业转载请注明出处。
16 */

```

分析：

根据假设"字符串中只含有小写字母",可以利用数组做散列计数每个小写字母出现的次数,然后依次对比两个统计数组是否对应相等即可.

- 方法一:分别计数法
- 方法二(方法一优化):加减计数法,[此方法受到了评论启发](#)

看到标签是“排序”，貌似完全没用到排序。

进阶Unicode未实现.

方法一:分别计数法

```

1  class Solution
2  {
3      public:
4          bool isAnagram(string s, string t)
5          {
6              int count_s[26] = {0,} ;
7              int count_t[26] = {0,} ;
8              int i          = 0      ;
9
10             if(s.size()!=t.size())
11             {
12                 return false;
13             }
14             for(i = 0;i < s.size();i++)
15             {
16                 count_s[s[i]-'a']++;
17             }
18             for(i = 0;i < t.size();i++)
19             {
20                 count_t[t[i]-'a']++;
21             }

```

```

22         for(i=0;i<26;i++)
23         {
24             if(count_s[i]!=count_t[i])
25             {
26                 return false;
27             }
28         }
29         return true;
30     }
31 };
32 /*
33 执行结果:
34 通过
35 显示详情
36 执行用时 :12 ms, 在所有 cpp 提交中击败了83.76% 的用户
37 内存消耗 :9.6 MB, 在所有 cpp 提交中击败了6.07%的用户
38 */

```

方法二:加减计数法

```

1  class Solution
2  {
3      public:
4          bool isAnagram(string s, string t)
5          {
6              int count[26] = {0,} ;
7              int i          = 0 ;
8
9              for(i = 0;i < s.size();i++)
10             {
11                 count[s[i]-'a']++;
12             }
13             for(i = 0;i < t.size();i++)
14             {
15                 count[t[i]-'a']--;
16             }
17             for(i=0;i<26;i++)
18             {
19                 if(count[i]!=0x00)
20                 {
21                     return false;
22                 }
23             }
24             return true;
25         }
26 };
27 /*
28 执行结果:
29 通过
30 显示详情
31 执行用时 :12 ms, 在所有 cpp 提交中击败了83.76% 的用户
32 内存消耗 :9.4 MB, 在所有 cpp 提交中击败了12.73%的用户
33 */

```

