# [字符串的排列](https://leetcode-cn.com/explore/interview/card/bytedance/242/string/1016/)

## 题目

给定两个字符串 s1 和 s2，写一个函数来判断 s2 是否包含 s1 的排列。

换句话说，第一个字符串的排列之一是第二个字符串的子串。

示例1:

输入: s1 = "ab" s2 = "eidbaooo"
输出: True
解释: s2 包含 s1 的排列之一 ("ba").

## 解题思路

1. 这道题，我们用到的算法是 **滑动窗口**
2. 首先字符串s1的排列的可能性应该是它的长度的阶乘，因为字符串长度可能为10000，所以找出所有排列情况是不太可能。
3. 我们可以转换思路，**不要关注排列的形式，而是关注排列中元素的数量关系**
4. 比如 aab，那么，转换为数量关系就是{a:2,b:1}，因为 S1 长度为 3，所以我们的窗口长度也为3
5. 如果我们在 S2 的找到了这样一个窗口符合出现 a 的次数是两个， b 是一个，那么 S2 就是包含 S1 的排列的

public boolean checkInclusion(String s1, String s2) {
 int len1 = s1.length();
 int len2 = s2.length();
 int[] c1 = new int[26];
 int[] c2 = new int[26];

 for (char c : s1.toCharArray()) {
 c1[c - 'a']++;
 }

 for (int i = 0; i < len2; i++) {
 if (i >= len1)
 --c2[s2.charAt(i - len1) - 'a'];//先把坐标查过的减掉
 c2[s2.charAt(i) - 'a']++;
 if (Arrays.equals(c1, c2))
 return true;
 }

 return false;
}