

0159. 至多包含两个不同字符的最长子串

👤 ITCharge 🕒 大约 1 分钟

- 标签：哈希表、字符串、滑动窗口
- 难度：中等

题目链接

- [0159. 至多包含两个不同字符的最长子串 - 力扣](#)

题目大意

给定一个字符串 s ，找出之多包含两个不同字符的最长子串 t ，并返回该子串的长度。

解题思路

使用滑动窗口来求解。

$left$, $right$ 指向字符串开始位置。

不断向右移动 $right$ 指针，使用 $count$ 变量来统计滑动窗口中共有多少个字符，以及使用哈希表来统计当前字符的频数。

当滑动窗口的字符多于 2 个时，向右移动 $left$ 指针，并减少哈希表中对应原 $left$ 指向字符的频数。

最后使用 max_count 来维护最长子串 t 的长度。

代码

```
import collections

class Solution:
    def lengthOfLongestSubstringTwoDistinct(self, s: str) -> int:
        max_count = 0
        k = 2
```

py

```
counts = collections.defaultdict(int)
count = 0
left, right = 0, 0
while right < len(s):
    if counts[s[right]] == 0:
        count += 1
    counts[s[right]] += 1
    right += 1
    if count > k:
        if counts[s[left]] == 1:
            count -= 1
        counts[s[left]] -= 1
        left += 1
    max_count = max(max_count, right - left)
return max_count
```