

567. Permutation in String

🕒 Created	@October 9, 2022 9:56 PM
📌 Difficulty	Medium
🔗 LC Url	https://leetcode.com/problems/permutation-in-string/
📌 Importance	
🏷️ Tag	NEET Sliding Window
📺 Video	

Given two strings `s1` and `s2`, return `true` if `s2` contains a permutation of `s1`, or `false` otherwise.

In other words, return `true` if one of `s1`'s permutations is the substring of `s2`.

Example 1:

```
Input: s1 = "ab", s2 = "eidbaooo"
Output: true
Explanation: s2 contains one permutation of s1 ("ba").
```

Example 2:

```
Input: s1 = "ab", s2 = "eidboao"
Output: false
```

Constraints:

- `1 <= s1.length, s2.length <= 104`
- `s1` and `s2` consist of lowercase English letters.

Solution

```

class Solution:
    def checkInclusion(self, s1: str, s2: str) -> bool:
        m1 = len(s1)
        m2 = len(s2)

        if m1 > m2:
            return False

        cnt1 = [0] * 26
        cnt2 = [0] * 26

        for i in range(m1):
            cnt1[ord(s1[i]) - ord('a')] += 1
            cnt2[ord(s2[i]) - ord('a')] += 1

        if cnt1 == cnt2:
            return True

        for i in range(m1, m2):
            cnt2[ord(s2[i - m1]) - ord('a')] -= 1
            cnt2[ord(s2[i]) - ord('a')] += 1
            if cnt1 == cnt2:
                return True

        return False

```