



## EXPERIMENT – 4

**Student Name:** arpit rana

**UID:** 22BCS10776

**Branch:** BE-CSE

**Section/Group:** NTPP-601-A

**Semester:** 6

**Date of Performance:** 11-02-2025

**Subject Name:** Advanced Programming 2 **Subject Code:** 22CSP-351

### 1. Aim:

- (a) **Find the Index of the First Occurrence in a String:** Given two strings needle and haystack, return the index of the first occurrence of needle in haystack, or -1 if needle is not part of haystack.
- (b) **Rotate String:** Given two strings 's' and 'goal', return true if and only if 's' can become 'goal' after some number of shifts on 's'.

### 2. Objectives:

- Find the Index of the First Occurrence in a String.
- Check if one string is a rotation of another.

### 3. Algorithm:

#### ➤ Find the Index of the First Occurrence in a String:

- Ensure that needle and haystack are compared as intended.
- The return I statement seems incorrect; it should be return i.
- Consider cases where needle is empty or longer than haystack.

#### ➤ Rotate String:

- If  $\text{len}(s) \neq \text{len}(\text{goal})$ , return False.
- Create  $s + s$  to include all possible rotations.
- If goal exists in  $s + s$ , return True; else, return False.

## 4. Implementation/Code:

### (a) Find the Index of the First Occurrence in a String:

```
class Solution:
    def strStr(self, haystack: str, needle: str) -> int:
        m = len(haystack)
        n = len(needle)

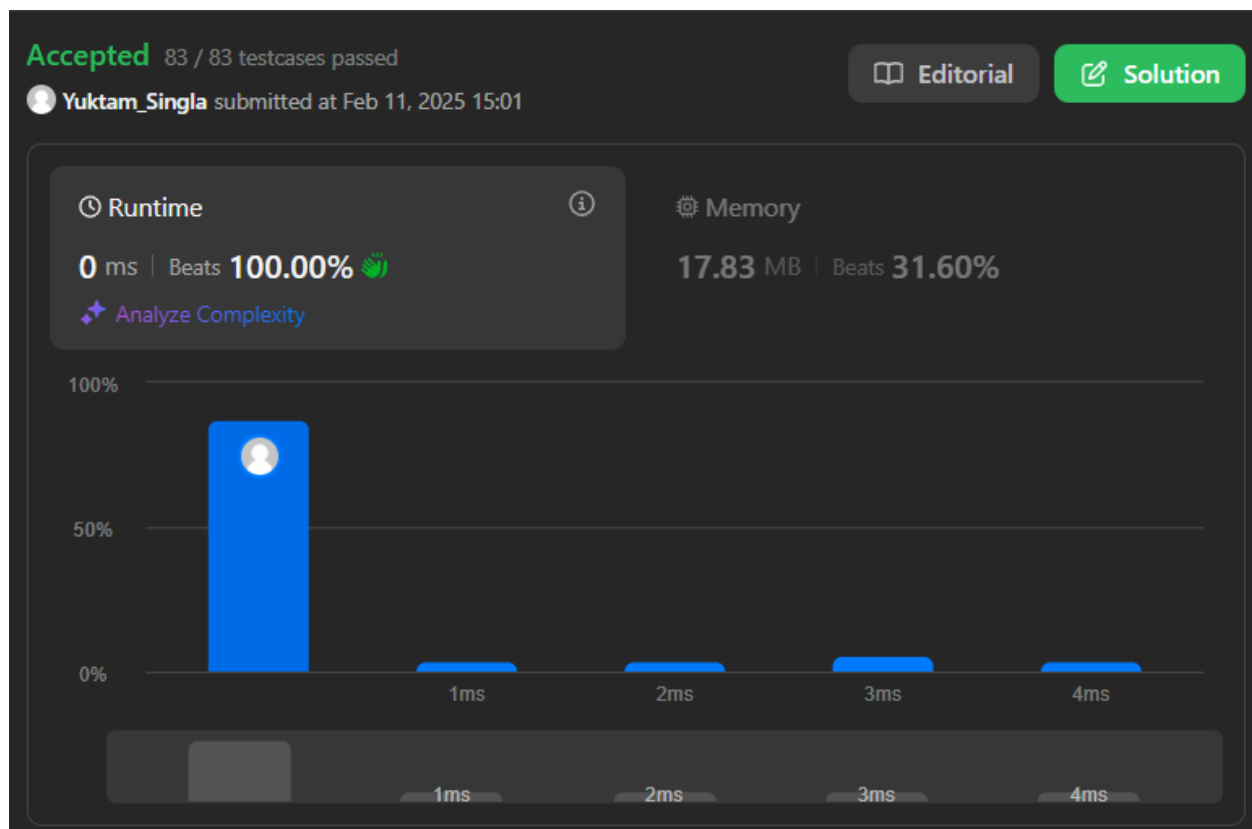
        for i in range(m - n + 1):
            if haystack[i:i + n] == needle:
                return i
```

### (b) Rotate String

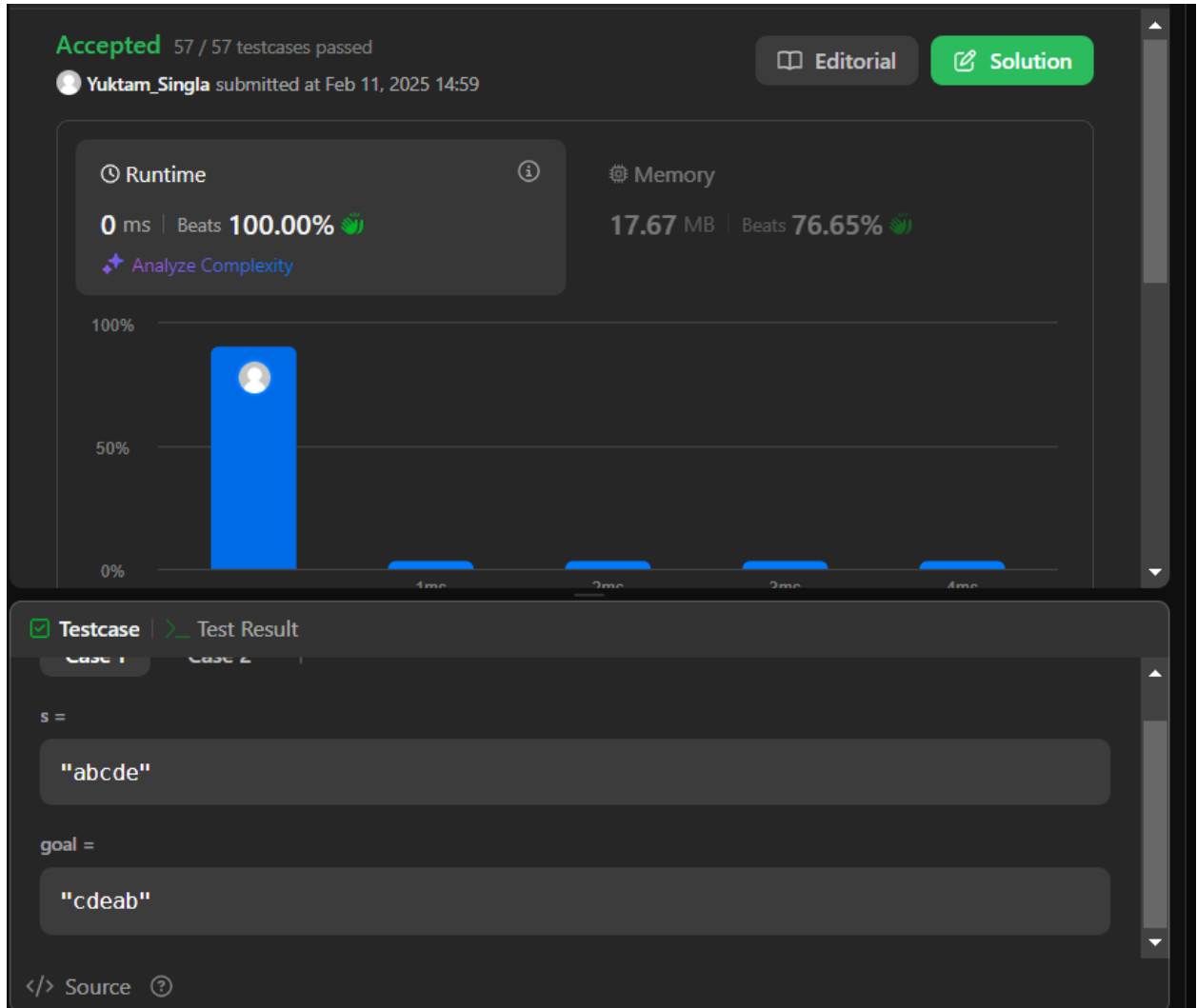
```
class Solution:
    def rotateString(self, s: str, goal: str) -> bool:
        return len(s) == len(goal) and goal in s + s
```

## 5. Output:

### (a) Find the Index of the First Occurrence in a String:



## (b) Rotate String:



## 6. Learning Outcomes:

- Understood how to check if one string is a rotation of another using string concatenation.
- Learned to use the find function to check for substring existence.
- Understood substring search using a sliding window.
- Learned to handle edge cases efficiently.
- Understood how to implement and test both functions in C++.