



Experiment-4

Student Name: Suryansh Gehlot

UID:22BCS10900

Branch: BE-CSE

Section/Group: KRG 2B

Semester: 6th

Date of Performance: 12-2-25

Subject Name: AP Lab-2

Subject Code: 22CSP-351

1. Aim: Find the Index of the First Occurrence in a String

2. Objective:

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.

Example 1:

Input: haystack = "sadbutsad", needle = "sad"

Output: 0

Explanation: "sad" occurs at index 0 and 6.

The first occurrence is at index 0, so we return 0.

3. Implementation/Code:

```
class Solution { public:
    int strStr(string haystack, string needle) {
        int n = haystack.size();    int m =
        needle.size();    if (m == 0) return 0;
        for (int i = 0; i <= n - m; ++i) {        if
        (haystack.substr(i, m) == needle) {
            return i;
        }
    }
}
```



```
    }  
  
    return -1;  
}  
};
```

4. Output

☒ Testcase | [>_ Test Result](#)

Accepted Runtime: 0 ms

• Case 1

• Case 2

Input

haystack =
"sadbutsad"

needle =
"sad"

Output

0

Expected

0

☒ Testcase | [>_ Test Result](#)

Accepted Runtime: 0 ms

• Case 1

• Case 2

Input

haystack =
"leetcode"

needle =
"leeto"

Output

-1

Expected

-1

5. Learning Outcome:

We Learn About the use of function.

We Learn About the use of vector.

We Learn About the use of indexing.

We Learn About the Calling For the function.

Question 2.

1. **Aim:** Missing Number

2. **Objective:**

Given an array nums containing n distinct numbers in the range $[0, n]$, return the only number in the range that is missing from the array.

Example 1:

Input: nums = [3,0,1]

Output: 2 Explanation:

$n = 3$ since there are 3 numbers, so all numbers are in the range $[0, 3]$. 2 is the missing number in the range since it does not appear in nums.

3. Implementation/Code:

```
class Solution { public:
    int missingNumber(vector<int>& nums) {
        int n = nums.size();

        int totalSum = n * (n + 1) / 2;

        int actualSum = 0;
        for (int num : nums) {
            actualSum += num;
        }

        return totalSum - actualSum;
    } };
```

4. Output

Saved

Testcase

>_ Test Result

Accepted

Runtime: 0 ms

• Case 1

• Case 2

Input

nums =
[3,0,1]

Output

2

Expected

2

Saved

Testcase

>_ Test Result

Accepted

Runtime: 0 ms

• Case 1

• Case 2

Input

nums =
[0,1]

Output

2

Expected

2

5. Learning Outcome:

- We Learn About the use of Running vector.
- We Learn About the use of vector.
- We Learn About the use of function and function calling.
- We learn About sum of n numbers.