

Aarush Wali

2022A6R002

Question number 4 :

The screenshot shows the LeetCode interface for the 'Two Sum' problem. The problem description on the left states: 'Given an array of integers `nums` and an integer `target`, return indices of the two numbers such that they add up to `target`. You may assume that each input would have **exactly one solution**, and you may not use the same element twice. You can return the answer in any order.'

Examples provided:

- Example 1: Input: `nums = [2,7,11,15]`, `target = 9`. Output: `[0,1]`. Explanation: Because `nums[0] + nums[1] == 9`, we return `[0,1]`.
- Example 2: Input: `nums = [3,2,4]`, `target = 6`. Output: `[1,2]`.
- Example 3: Input: `nums = [3,3]`, `target = 6`. Output: `[0,1]`.

The C++ code on the right implements a solution using a nested loop:

```
class Solution {
public:
    vector<int> twoSum(vector<int>& nums, int target) {
        int s,k;
        for(int i=0;i<nums.size();i++){
            for(int j=i+1;j<nums.size();j++){
                if((nums[i]+nums[j])==target){
                    s=i;
                    k=j;
                    break;
                }
            }
        }
        return {s,k};
    }
};
```

The test result shows 'Accepted' with a runtime of 3 ms. The input for the test case is `nums = [2,7,11,15]`.

This screenshot shows the same 'Two Sum' problem page but with performance metrics and a different C++ solution. The 'All Submissions' tab is active, showing a submission that was 'Accepted' and submitted at Jul 17, 2024 11:51.

Performance metrics:

- Runtime: 46 ms | Beats 41.45%
- Memory: 12.92 MB | Beats 55.51%

The C++ code on the right is a more concise version of the solution:

```
public:
    vector<int> twoSum(vector<int>& nums, int target) {
        int s,k;
        for(int i=0;i<nums.size();i++){
            for(int j=i+1;j<nums.size();j++){
                if((nums[i]+nums[j])==target){
                    s=i;
                    k=j;
                    break;
                }
            }
        }
        return {s,k};
    }
```

The test result shows 'Accepted' with a runtime of 3 ms. The input for the test case is `nums = [2,7,11,15]`.