**Experiment - 5**

|  |  |
| --- | --- |
| **Student Name: Aditi Mourya** | **UID: 22BCS11624** |
| **Branch: CSE** | **Section: TPP-IOT-638-A** |
| **Semester: 6th** | **DOP:19/02/2025** |
| **Subject: Advanced Programming-II** | **Subject Code:22CSP-351** |

**Problem-1: Sort Colors**

Given an array nums with n objects colored red, white, or blue, sort them in-place so that objects of the same color are adjacent, with the colors in the order red, white, and blue. We will use the integers 0, 1, and 2 to represent the color red, white, and blue, respectively. You must solve this problem without using the library's sort function.

Example 1:

Input: nums = [2,0,2,1,1,0] Output: [0,0,1,1,2,2]

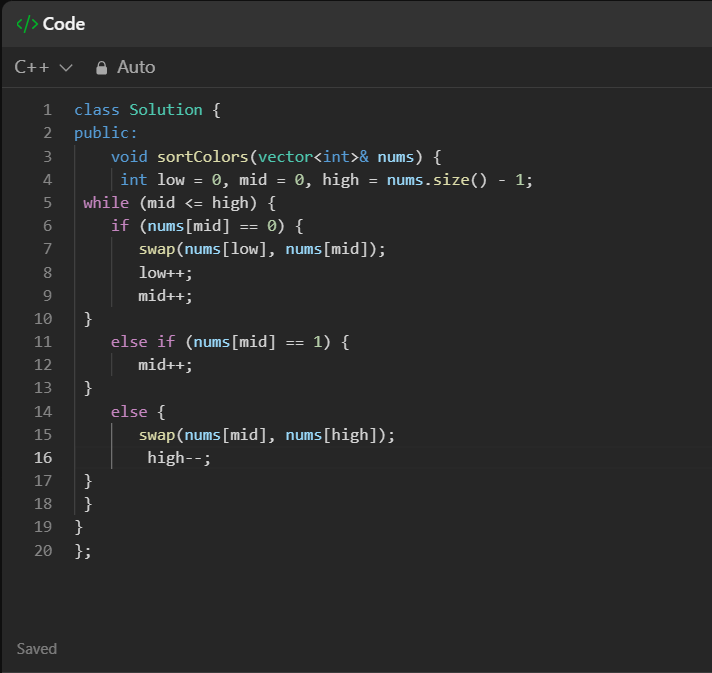
Example 2:

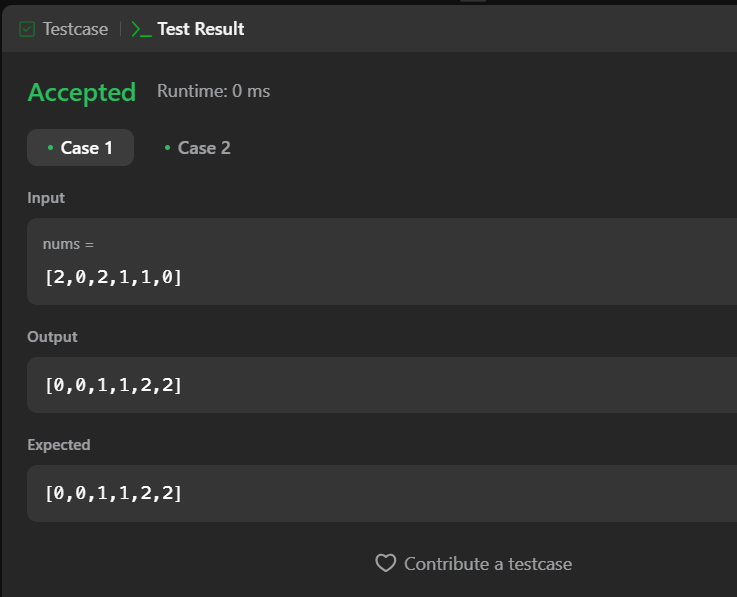
Input: nums = [2,0,1] Output: [0,1,2]

Constraints:

n == nums.length 1 <= n <= 300 nums[i] is either 0, 1, or 2.

Follow up: Could you come up with a one-pass algorithm using only constant extra space?





**Problem-2: Kth Largest Element in an Array**

Given an integer array nums and an integer k, return the kth largest element in the array. Note that it is the kth largest element in the sorted order, not the kth distinct element. Can you solve it without sorting?

Example 1:

Input: nums = [3,2,1,5,6,4], k = 2 Output: 5

Example 2:

Input: nums = [3,2,3,1,2,4,5,5,6], k = 4 Output: 4

Constraints:

1 <= k <= nums.length <= 105 -104 <= nums[i] <= 104

