Assignment 2

Computing Lab (CS702)

Qns 1. Given an array of 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 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.

Test Cases:

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

Qns 2. Given an array of prices where prices[i] is the price of a given stock on the i^{th} day. You want to maximize your profit by choosing a single day to buy one stock and choosing a different day in the future to sell that stock. Return the maximum profit you can achieve from this transaction. If you cannot make a profit, return 0.

Test Cases:

1. Input: prices = [7,1,5,3,6,4]
   Output: 5

(Explanation: Buy on day 2 (price = 1) and sell on day 5 (price = 6), profit = 6-1 = 5. Note that buying on day 2 and selling on day 1 is not allowed because you must buy before you sell.)

Qns 3. Given an integer array of size n, find all elements that appear more than ⌊ n/3 ⌋ times.

Test Cases:

1. Input: nums = [3,2,3]
   Output: [3]
2. Input: nums = [1]
   Output: [1]
3. Input: nums = [1,2]
   Output: [1,2]

Qns 4. Given a sorted array consisting of only integers where every element appears exactly twice, except for one element which appears exactly once. Return the single element that occurs only once. The solution must run in O(log n) time and O(1) space.

Test Cases:

1. Input: nums = [1,1,2,3,3,4,4,8,8]
   Output: 2
2. Input: nums = [3,3,7,7,10,11,11]
   Output: 10

Qns 5. Given an array of integers nums sorted in non-decreasing order, find the starting and ending position of a given target value. If the target is not found in the array, return [-1, -1].

The algorithm must have an O(log n) runtime complexity.

Test Cases:

1. Input: nums = [5,7,7,8,8,10], target = 8
   Output: [3,4]
2. Input: nums = [5,7,7,8,8,10], target = 6
   Output: [-1,-1]
3. Input: nums = [], target = 0
   Output: [-1,-1]

Qns 6. Search in Rotated Sorted Array. There is an integer array nums sorted in ascending order (with distinct values). The array is rotated at an unknown pivot index k (0 <= k < n). Given the array nums and an integer target, return the index of target if it is in nums, or -1 if it is not. You must write an algorithm with O(log n) runtime complexity.

Test Cases:

1. Input: nums = [4,5,6,7,0,1,2], target = 0
   Output: 4
2. Input: nums = [4,5,6,7,0,1,2], target = 3
   Output: -1

Qns 7. Median of Two Sorted Arrays. Given two sorted arrays, nums1 and nums2, of size m and n, return the median of the two sorted arrays. The overall runtime complexity should be O(log (m+n)).

Test Cases:

1. Input: nums1 = [1,3], nums2 = [2]
   Output: 2.0
2. Input: nums1 = [1,2], nums2 = [3,4]
   Output: 2.5

Qns 8. K^{th} Smallest Element in a Sorted Matrix. Given an n x n matrix where each row and column is sorted in ascending order, return the kth smallest element in the matrix. You must find a solution with a memory complexity better than $O(n^2)$.

Test Cases:

1. Input:
   matrix = [[1,5,9],
   [10,11,13],
   [12,13,15]], k = 8
   Output: 13
2. Input:
   matrix = [[-5]], k = 1
   Output: -5

Qns 9. Given an integer array nums, find the contiguous subarray with the most significant sum, and return its sum.

Test Cases:

1. Input: [-2, 1, -3, 4, -1, 2, 1, -5, 4]

Output: 6
2. Input: [1]
   Output: 1
3. Input: [5, 4, -1, 7, 8]
   Output: 23

Qns 10. A peak element is an element that is strictly greater than its neighbors. Given a 0-indexed integer array nums, find a peak element and return its index. If the array contains multiple peaks, return the index of any of the peaks. Assume that nums[-1] = nums[n] = -. The algorithm should run in O(log n) time.

Test Cases:

1. Input: [1,2,3,1]
   Output: 2
2. Input: [1,2,1,3,5,6,4]
   Output: 5