ARRAYS – QUESTIONS

Question 1: Given an integer array nums, return true if any value appears at least twice in the array, and return false if every element is distinct.

Example 1: Input: nums = [1, 2, 3, 1] Output: true

Example 2: Input: nums = [1, 2, 3, 4] Output: false

Example 3: Input: nums = [1, 1, 1, 3, 3, 4, 3, 2, 4, 2] Output: true

Constraints: • 1 <= nums . lengtth <= 105

• -109 <= nums [i] <= 109

Question 2: There is an integer array nums sorted in ascending order (with distinct values). Prior to being passed to your function, nums is possibly rotated at an unknown pivot index k (1 <= k < nums.length) such that the resulting array is [nums[k], nums[k+1], ..., nums[n-1], nums[0], nums[1], ..., nums[k-1]] (0-indexed). For example, [0,1,2,4,5,6,7] might be rotated at pivot index 3 and become [4,5,6,7,0,1,2].

Given the array nums after the possible rotation and an integer target, return the index of target if it is in nums, or -1 if it is not in nums. You must write an algorithm with O(log n) runtime complexity.

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

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

Example 3: Input: nums = [1], target = 0 Output: -1

Constraints: • 1 <= nums . lengtth <= 5000

• -104 <= nums [i] <= 104

• All values of nums are unique.

• nums is an ascending array that is possibly rotated.

• -104 <= target <= 104

Question 3: You are given an array prices where prices[i] is the price of a given stock on the i th day. Return the maximum profit you can achieve from this transaction. If you cannot achieve any profit, return 0.

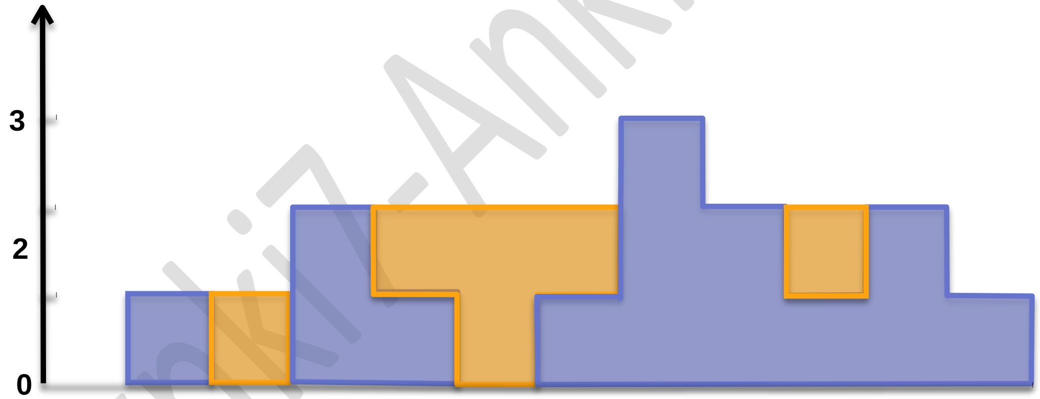
Example 1: Input: prices = [7, 1, 5, 3, 6, 4] Output: 5 Explanation: Buy on day 2 (price = 1) and sell on day 5 (price = 6), profit = 6-1 = 5. Note that buying on day 2 and selling on day 1 is not allowed because you must buy before you sell.

Example 2: Input: Prices = [7, 6, 4, 3, 1] Output: 0 Explanation: In this case, no transactions are done and the max profit = 0.

Constraints: • 1 <= prices . length <= 105

• 0 <= prices [i] <= 104

Question 4: Given n non-negative integers representing an elevation map where the width of each bar is 1, compute how much water it can trap after raining.



Example 1:

Input: height = [0, 1, 0, 2, 1, 0, 1, 3, 2, 1, 2, 1]

Output: 6

Explanation: The above elevation map (black section) is represented by array [0,1,0,2,1,0,1,3,2,1,2,1]. In this case, 6 units of rain water (blue section)are being trapped.

Example 2:

Input: height = [4, 2, 0, 3, 2, 5]

Output: 9

Constraints:

- n == height . length
- 1 <= n <= 2 * 10^4
- 0 <= height [i] < = 10^5

Question 5: Given an integer array nums, return all the triplets [nums[i], nums[j], nums[k]] such that i != j, i != k, and j != k, and nums[i] + nums[j] + nums[k] == 0. Notice that the solution set must not contain duplicate triplets.

Example 1: Input: nums = [-1, 0, 1, 2, -1, -4] Output: [[-1, -1, 2] , [-1, 0, 1]]

Example 2: Input: nums = [ ] Output: [ ]

Example 3: Input: nums = [0] Output: [ ]

Constraints: • 0 <= nums . length <= 3000

• -105 <= nums [i] <= 105