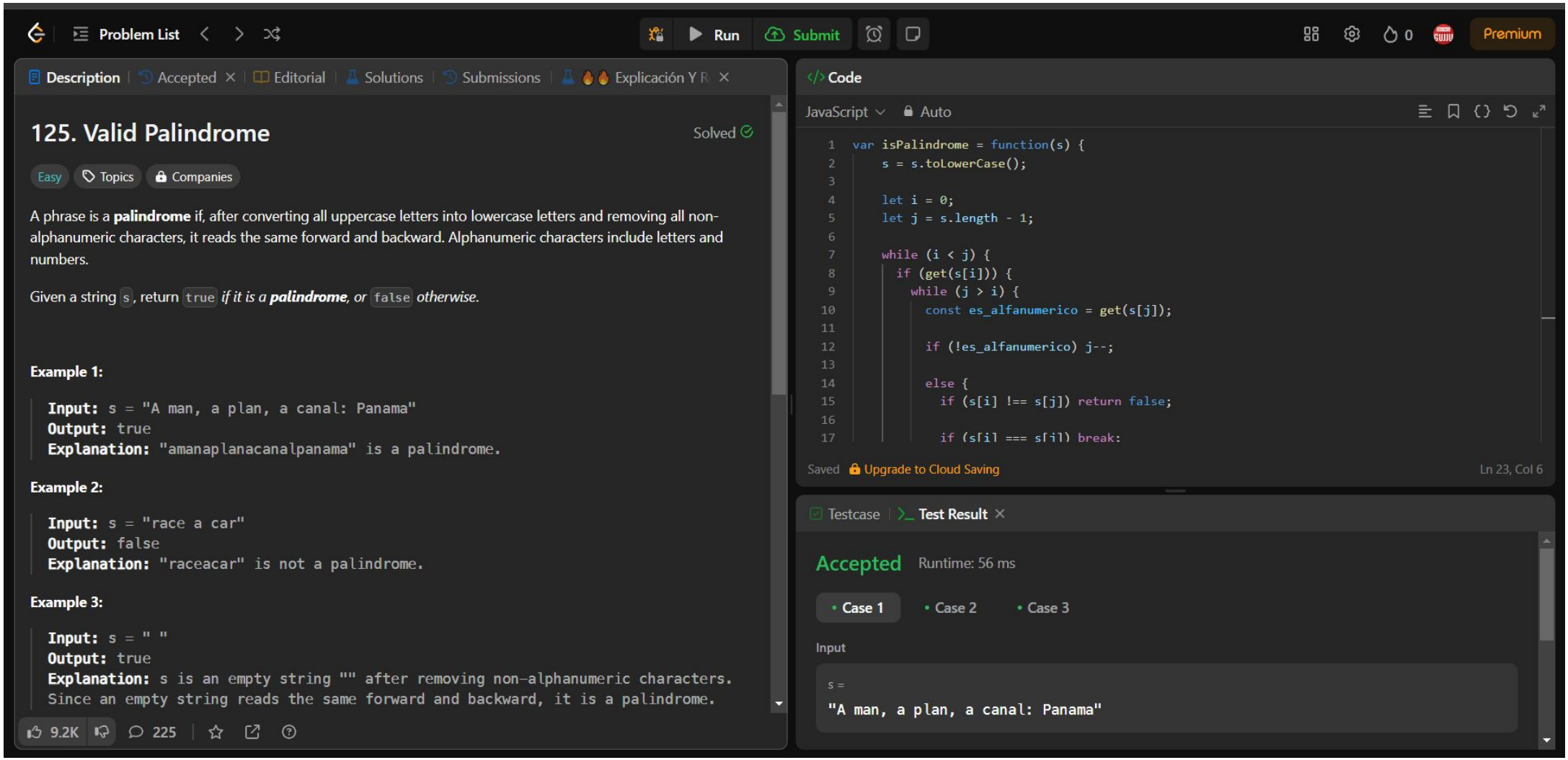
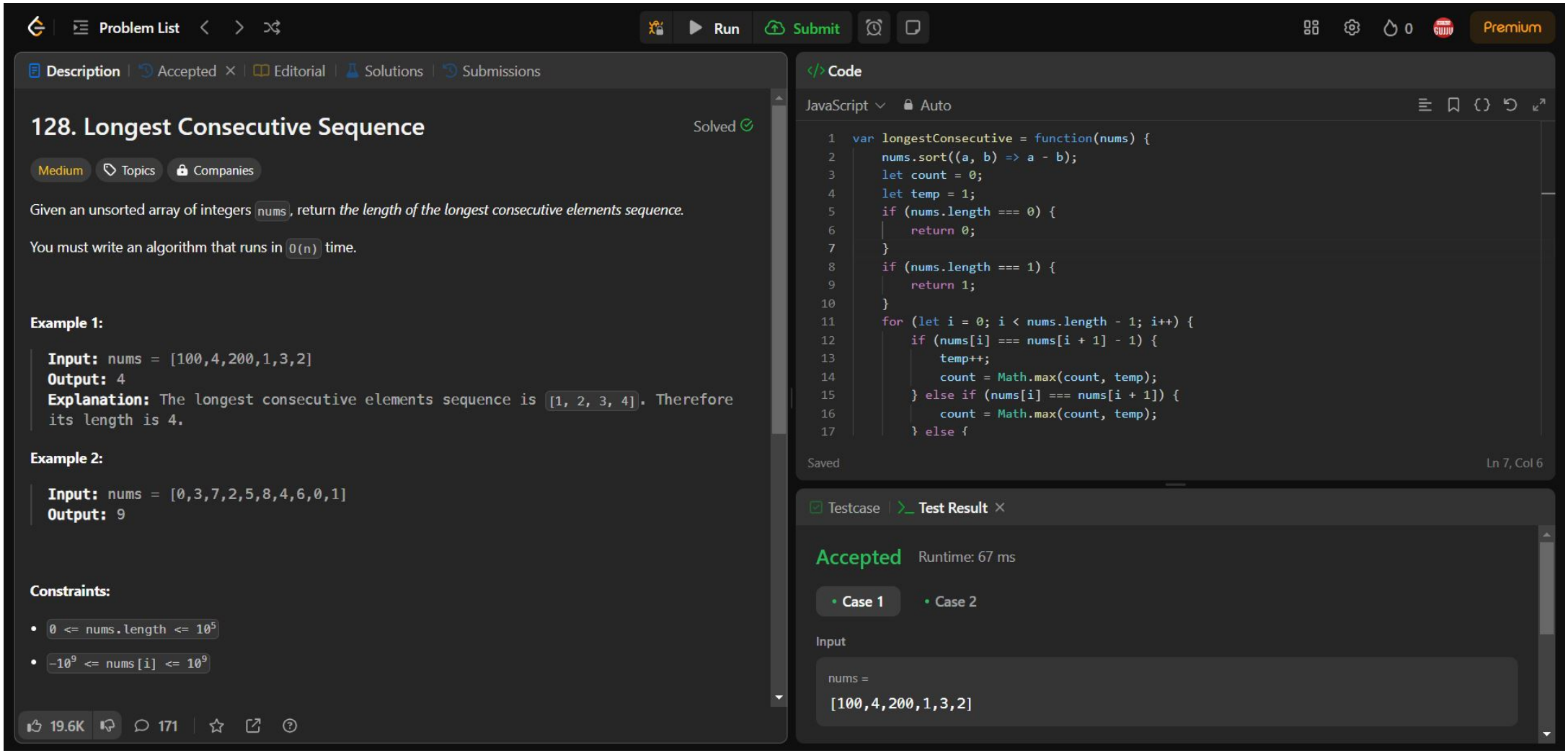
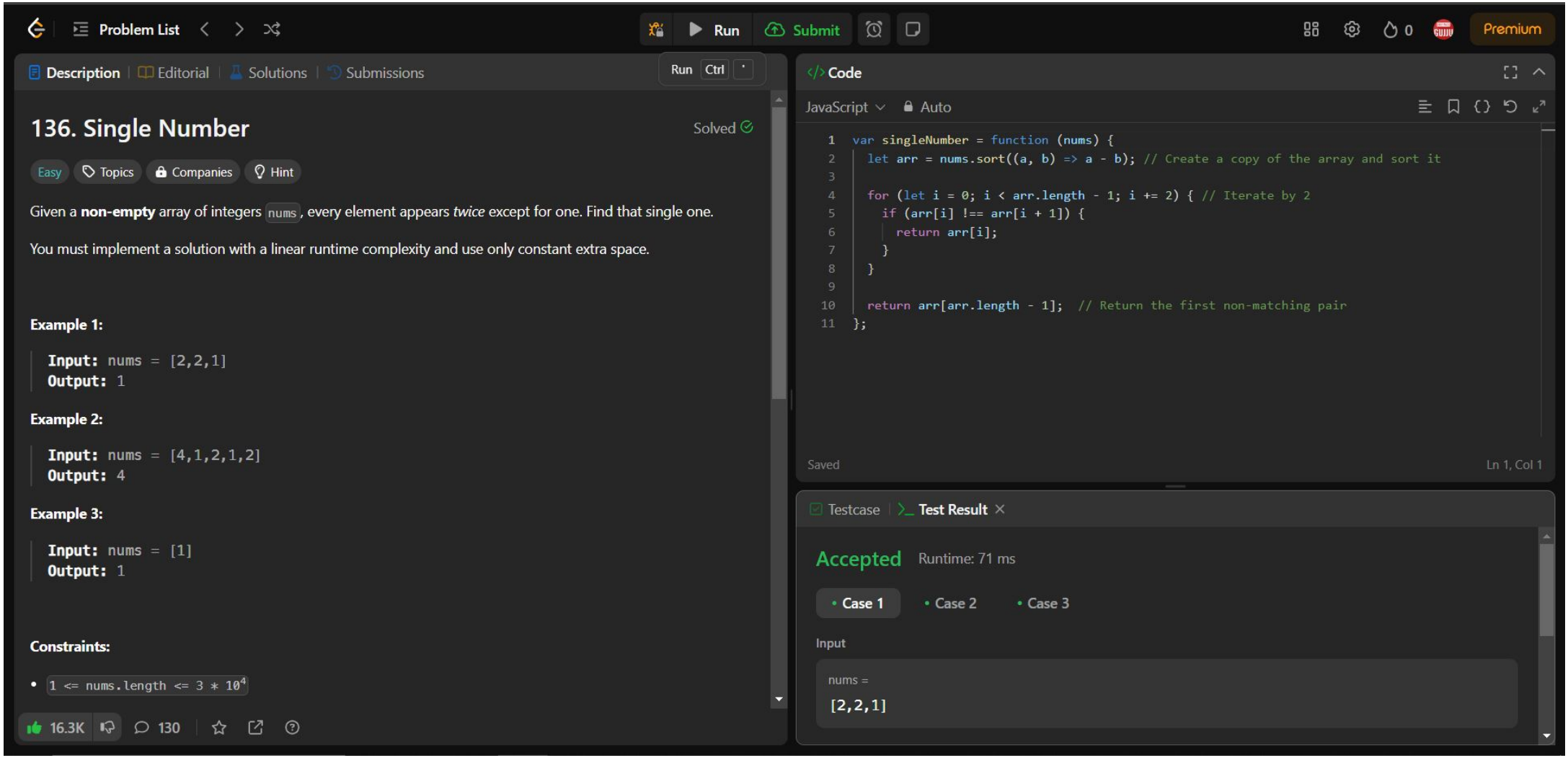
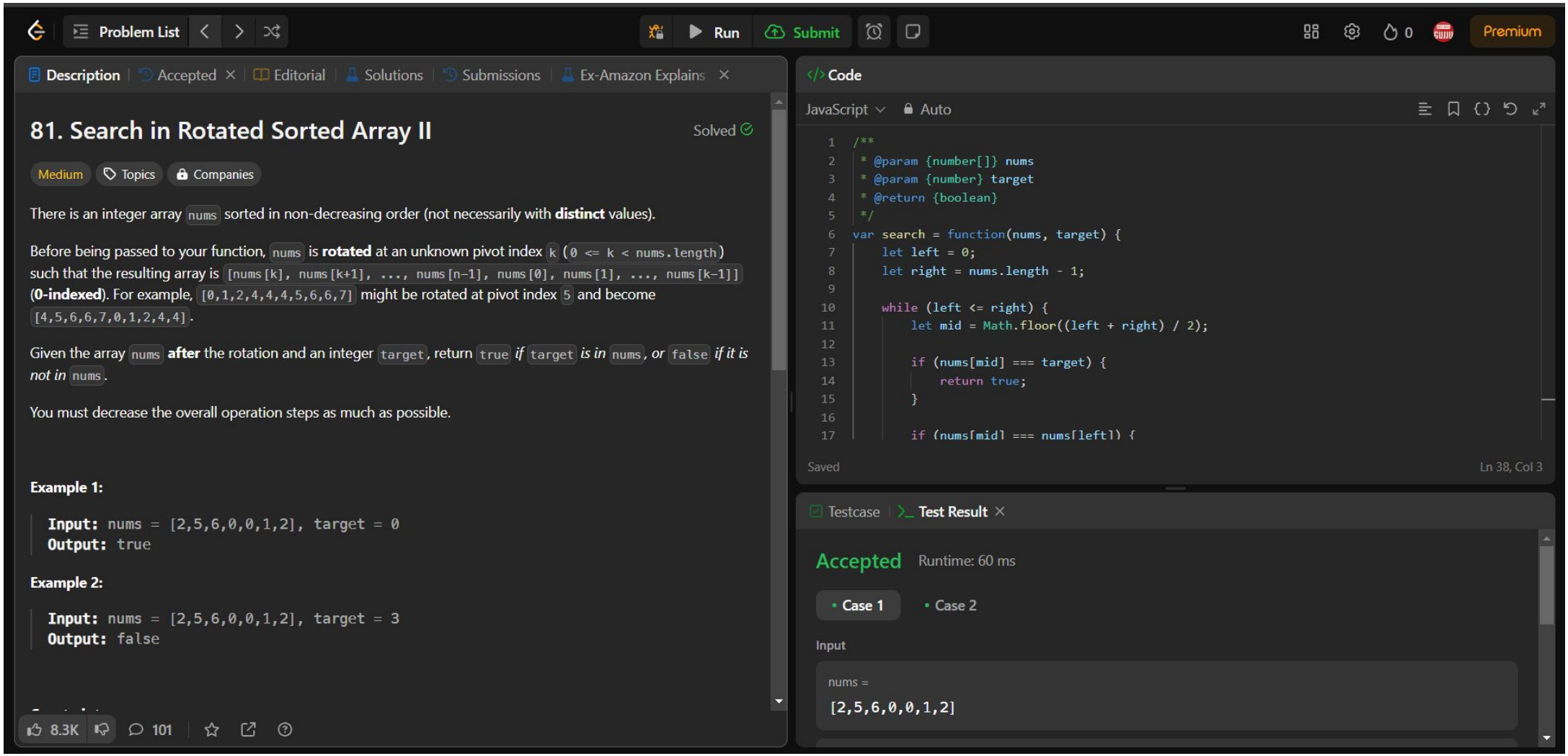
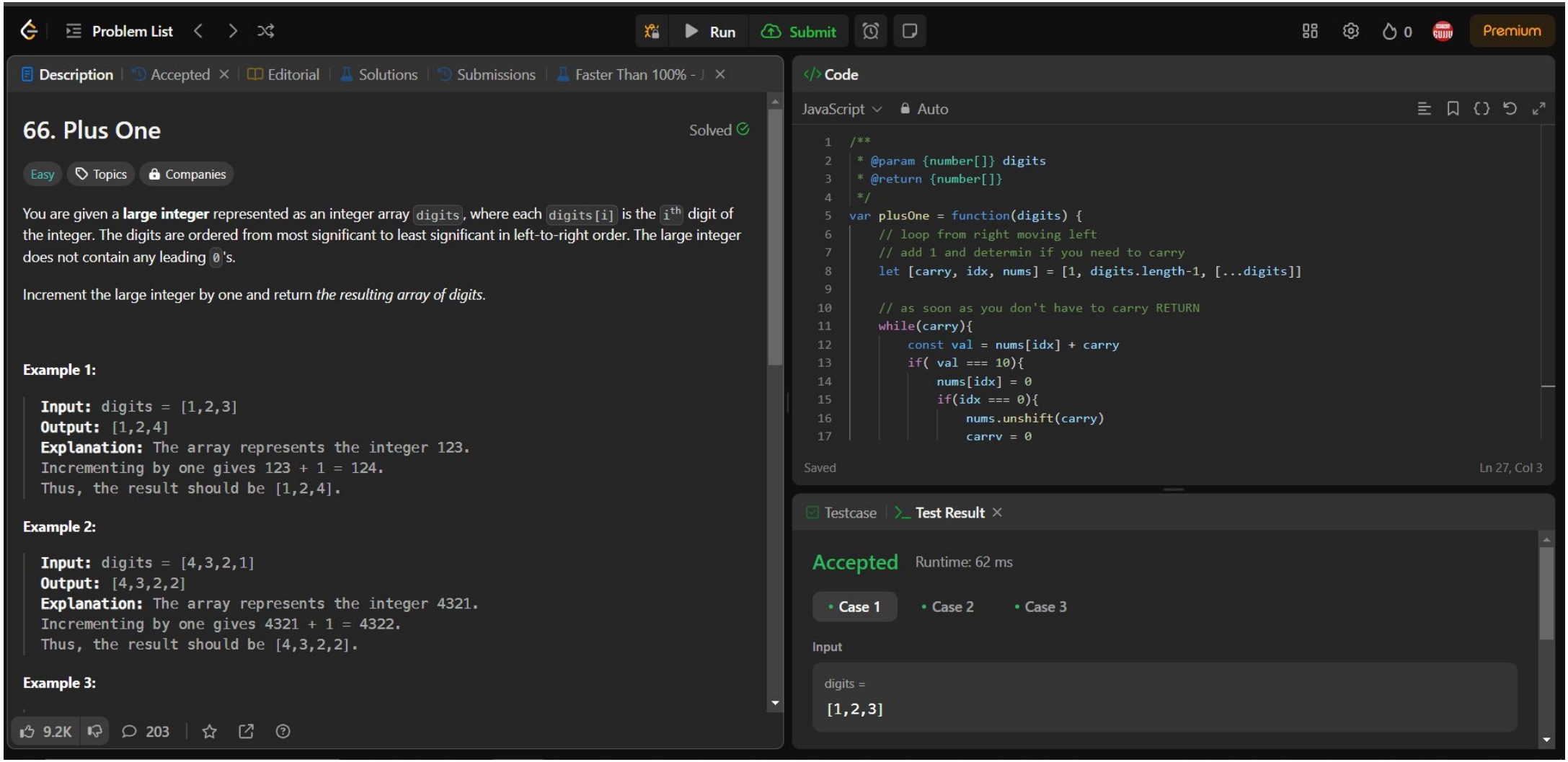

Problem List
Run
Submit
0
Premium
Description
Accepted
Editorial
Solutions
Submissions
Explicación Y R
125. Valid Palindrome
Solved
Easy
Topics
Companies
A phrase is a palindrome if, after converting all uppercase letters into lowercase letters and removing all non-alphanumeric characters, it reads the same forward and backward. Alphanumeric characters include letters and numbers.
Given a string s, return true if it is a palindrome, or false otherwise.
Example 1:
Input: s = "A man, a plan, a canal: Panama"
Output: true
Explanation: "amanaplanacanalpanama" is a palindrome.
Example 2:
Input: s = "race a car"
Output: false
Explanation: "raceacar" is not a palindrome.
Example 3:
Input: s = " "
Output: true
Explanation: s is an empty string "" after removing non-alphanumeric characters.
Since an empty string reads the same forward and backward, it is a palindrome.
9.2K
225
Code
JavaScript
Auto
var isPalindrome = function(s) {
s = s.toLowerCase();
let i = 0;
let j = s.length - 1;
while (i < j) {
if (get(s[i])) {
while (j > i) {
const es_alfanumerico = get(s[j]);
if (!es_alfanumerico) j--;
else {
if (s[i] !== s[j]) return false;
if (s[i] === s[i]) break;
Saved
Upgrade to Cloud Saving
Ln 23, Col 6
Testcase
Test Result
Accepted
Runtime: 56 ms
Case 1
Case 2
Case 3
Input
s =
"A man, a plan, a canal: Panama"


Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
128. Longest Consecutive Sequence
Solved
Medium
Topics
Companies
Given an unsorted array of integers nums, return the length of the longest consecutive elements sequence.
You must write an algorithm that runs in O(n) time.
Example 1:
Input: nums = [100,4,200,1,3,2]
Output: 4
Explanation: The longest consecutive elements sequence is [1, 2, 3, 4]. Therefore its length is 4.
Example 2:
Input: nums = [0,3,7,2,5,8,4,6,0,1]
Output: 9
Constraints:
0 <= nums.length <= 10^5
-10^9 <= nums[i] <= 10^9
19.6K
171
Code
JavaScript
Auto
var longestConsecutive = function(nums) {
nums.sort((a, b) => a - b);
let count = 0;
let temp = 1;
if (nums.length === 0) {
return 0;
}
if (nums.length === 1) {
return 1;
}
for (let i = 0; i < nums.length - 1; i++) {
if (nums[i] === nums[i + 1] - 1) {
temp++;
count = Math.max(count, temp);
} else if (nums[i] === nums[i + 1]) {
count = Math.max(count, temp);
} else {
Saved
Ln 7, Col 6
Testcase
Test Result
Accepted
Runtime: 67 ms
Case 1
Case 2
Input
nums =
[100,4,200,1,3,2]


Problem List
Run
Submit
Premium
Description
Editorial
Solutions
Submissions
Run Ctrl '
136. Single Number
Solved
Easy
Topics
Companies
Hint
Given a non-empty array of integers nums, every element appears twice except for one. Find that single one.
You must implement a solution with a linear runtime complexity and use only constant extra space.
Example 1:
Input: nums = [2,2,1]
Output: 1
Example 2:
Input: nums = [4,1,2,1,2]
Output: 4
Example 3:
Input: nums = [1]
Output: 1
Constraints:
1 <= nums.length <= 3 * 10^4
16.3K
130
Code
JavaScript
Auto
var singleNumber = function (nums) {
let arr = nums.sort((a, b) => a - b); // Create a copy of the array and sort it
for (let i = 0; i < arr.length - 1; i += 2) { // Iterate by 2
if (arr[i] !== arr[i + 1]) {
return arr[i];
}
}
return arr[arr.length - 1]; // Return the first non-matching pair
};
Saved
Ln 1, Col 1
Testcase
Test Result
Accepted
Runtime: 71 ms
Case 1
Case 2
Case 3
Input
nums =
[2,2,1]


Problem List
Run
Submit
0
Premium
Description
Accepted
Editorial
Solutions
Submissions
Ex-Amazon Explains
81. Search in Rotated Sorted Array II
Solved
Medium
Topics
Companies
There is an integer array nums sorted in non-decreasing order (not necessarily with distinct values).
Before being passed to your function, nums is rotated at an unknown pivot index k (0 <= 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, [0,1,2,4,4,4,5,6,6,7] might be rotated at pivot index 5 and become [4,5,6,6,7,0,1,2,4,4].
Given the array nums after the rotation and an integer target, return true if target is in nums, or false if it is not in nums.
You must decrease the overall operation steps as much as possible.
Example 1:
Input: nums = [2,5,6,0,0,1,2], target = 0
Output: true
Example 2:
Input: nums = [2,5,6,0,0,1,2], target = 3
Output: false
8.3K
101
Code
JavaScript
Auto
/**
* @param {number[]} nums
* @param {number} target
* @return {boolean}
*/
var search = function(nums, target) {
let left = 0;
let right = nums.length - 1;
while (left <= right) {
let mid = Math.floor((left + right) / 2);
if (nums[mid] === target) {
return true;
}
if (nums[mid] === nums[left]) {
Saved
Ln 38, Col 3
Testcase
Test Result
Accepted
Runtime: 60 ms
Case 1
Case 2
Input
nums =
[2,5,6,0,0,1,2]


Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
Faster Than 100% -
66. Plus One
Solved
Easy
Topics
Companies
You are given a large integer represented as an integer array digits, where each digits[i] is the i^th digit of the integer. The digits are ordered from most significant to least significant in left-to-right order. The large integer does not contain any leading 0's.
Increment the large integer by one and return the resulting array of digits.
Example 1:
Input: digits = [1,2,3]
Output: [1,2,4]
Explanation: The array represents the integer 123.
Incrementing by one gives 123 + 1 = 124.
Thus, the result should be [1,2,4].
Example 2:
Input: digits = [4,3,2,1]
Output: [4,3,2,2]
Explanation: The array represents the integer 4321.
Incrementing by one gives 4321 + 1 = 4322.
Thus, the result should be [4,3,2,2].
Example 3:
9.2K
203
Code
JavaScript
Auto
/**
* @param {number[]} digits
* @return {number[]}
*/
var plusOne = function(digits) {
// loop from right moving left
// add 1 and determin if you need to carry
let [carry, idx, nums] = [1, digits.length-1, [...digits]]
// as soon as you don't have to carry RETURN
while(carry){
const val = nums[idx] + carry
if(val === 10){
nums[idx] = 0
if(idx === 0){
nums.unshift(carry)
carry = 0
Saved
Ln 27, Col 3
Testcase
Test Result
Accepted
Runtime: 62 ms
Case 1
Case 2
Case 3
Input
digits =
[1,2,3]