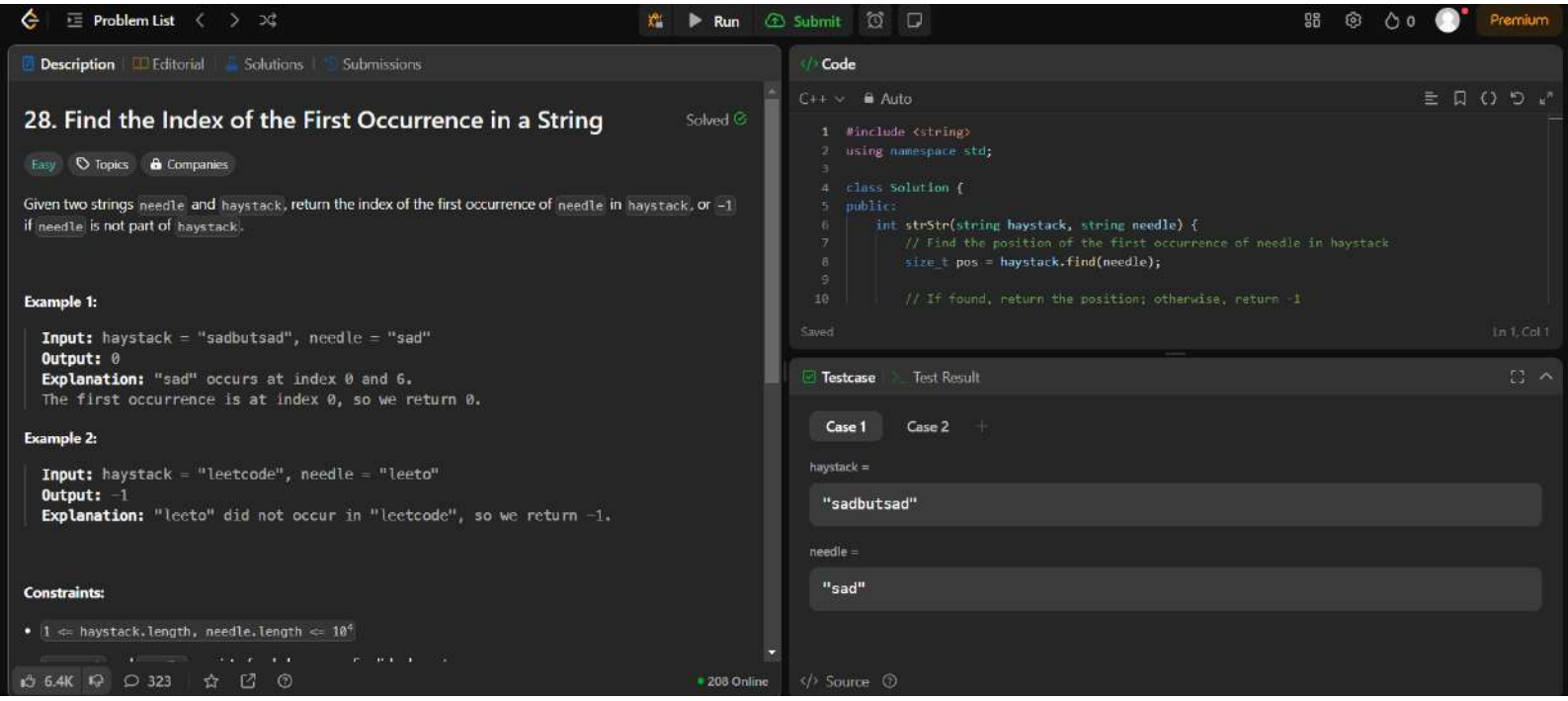
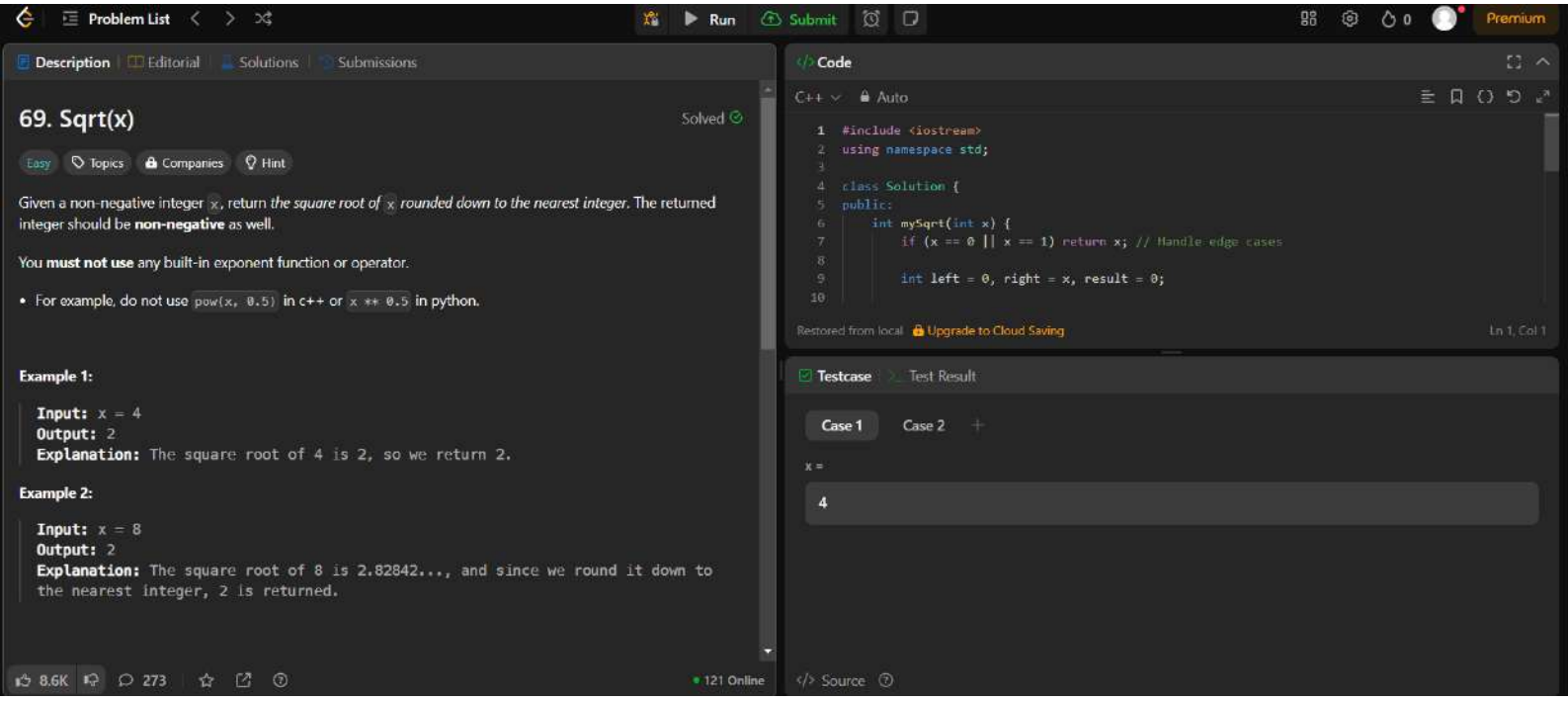
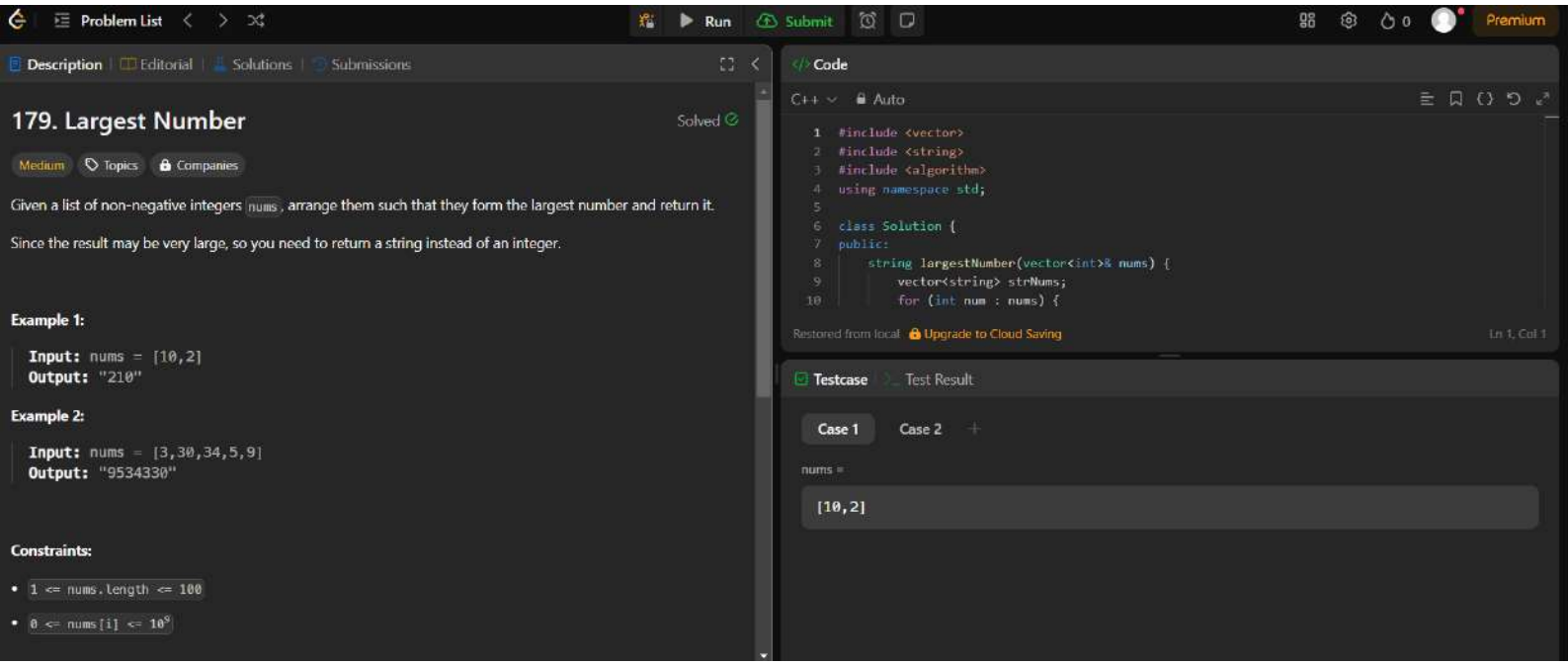
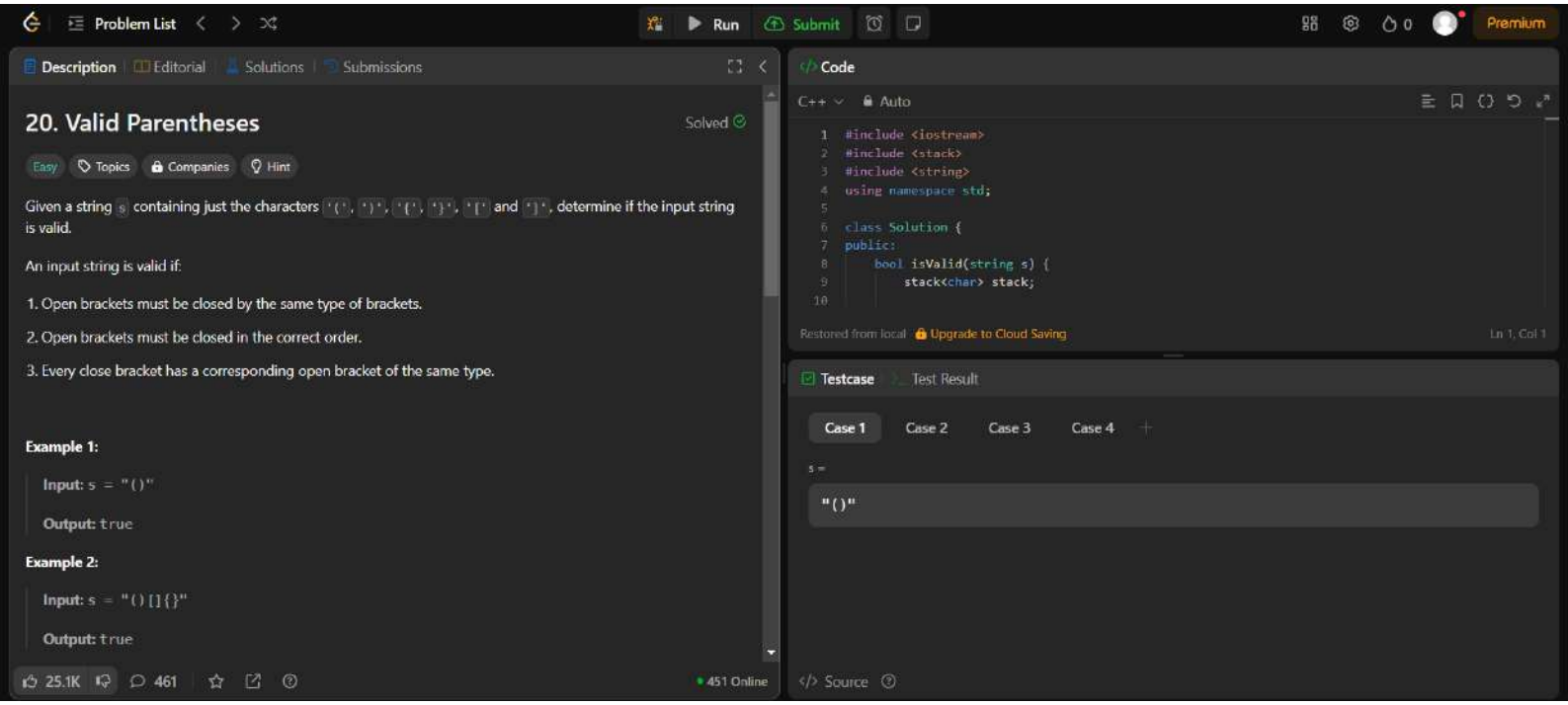

Problem List
Run
Submit
Premium
Description
Editorial
Solutions
Submissions
28. Find the Index of the First Occurrence in a String
Solved
Easy
Topics
Companies
Given two strings needle and haystack, return the index of the first occurrence of needle in haystack, or -1 if needle is not part of haystack.
Example 1:
Input: haystack = "sadbutsad", needle = "sad"
Output: 0
Explanation: "sad" occurs at index 0 and 6.
The first occurrence is at index 0, so we return 0.
Example 2:
Input: haystack = "leetcode", needle = "leeto"
Output: -1
Explanation: "leeto" did not occur in "leetcode", so we return -1.
Constraints:
1 <= haystack.length, needle.length <= 10^4
6.4K
323
208 Online
Code
C++
Auto
#include <string>
using namespace std;
class Solution {
public:
int strStr(string haystack, string needle) {
// Find the position of the first occurrence of needle in haystack
size_t pos = haystack.find(needle);
// If found, return the position; otherwise, return -1
Saved
Ln 1, Col 1
Testcase
Test Result
Case 1
Case 2
haystack =
"sadbutsad"
needle =
"sad"
Source


Problem List
Run
Submit
Premium
Description
Editorial
Solutions
Submissions
69. Sqrt(x)
Solved
Easy
Topics
Companies
Hint
Given a non-negative integer x, return the square root of x rounded down to the nearest integer. The returned integer should be non-negative as well.
You must not use any built-in exponent function or operator.
For example, do not use pow(x, 0.5) in c++ or x ** 0.5 in python.
Example 1:
Input: x = 4
Output: 2
Explanation: The square root of 4 is 2, so we return 2.
Example 2:
Input: x = 8
Output: 2
Explanation: The square root of 8 is 2.82842..., and since we round it down to the nearest integer, 2 is returned.
8.6K
273
121 Online
Code
C++
Auto
#include <iostream>
using namespace std;
class Solution {
public:
int mySqrt(int x) {
if (x == 0 || x == 1) return x; // Handle edge cases
int left = 0, right = x, result = 0;
Restored from local
Upgrade to Cloud Saving
Ln 1, Col 1
Testcase
Test Result
Case 1
Case 2
x =
4
Source


Problem List
Run
Submit
Premium
Description
Editorial
Solutions
Submissions
179. Largest Number
Solved
Medium
Topics
Companies
Given a list of non-negative integers nums, arrange them such that they form the largest number and return it.
Since the result may be very large, so you need to return a string instead of an integer.
Example 1:
Input: nums = [10,2]
Output: "210"
Example 2:
Input: nums = [3,30,34,5,9]
Output: "9534330"
Constraints:
1 <= nums.length <= 100
0 <= nums[i] <= 10^9
Code
C++
Auto
#include <vector>
#include <string>
#include <algorithm>
using namespace std;
class Solution {
public:
string largestNumber(vector<int>& nums) {
vector<string> strNums;
for (int num : nums) {
Restored from local
Upgrade to Cloud Saving
Ln 1, Col 1
Testcase
Test Result
Case 1
Case 2
nums =
[10,2]


Problem List
Run
Submit
Premium
Description
Editorial
Solutions
Submissions
20. Valid Parentheses
Solved
Easy
Topics
Companies
Hint
Given a string s containing just the characters '(', ')', '{', '}', '[' and ']', determine if the input string is valid.
An input string is valid if:
1. Open brackets must be closed by the same type of brackets.
2. Open brackets must be closed in the correct order.
3. Every close bracket has a corresponding open bracket of the same type.
Example 1:
Input: s = "()"
Output: true
Example 2:
Input: s = "()[]{}"
Output: true
25.1K
461
451 Online
Code
C++
Auto
#include <iostream>
#include <stack>
#include <string>
using namespace std;
class Solution {
public:
bool isValid(string s) {
stack<char> stack;
Restored from local
Upgrade to Cloud Saving
Ln 1, Col 1
Testcase
Test Result
Case 1
Case 2
Case 3
Case 4
s =
"()"
Source

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

21. Merge Two Sorted Lists

Solved

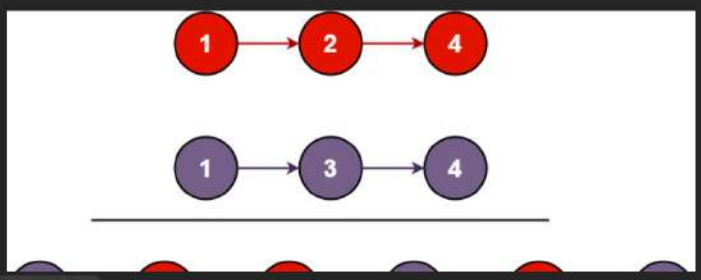
Easy | Topics | Companies

You are given the heads of two sorted linked lists `list1` and `list2`.

Merge the two lists into one **sorted** list. The list should be made by splicing together the nodes of the first two lists.

Return *the head of the merged linked list.*

Example 1:



22.7K | 397 | 441 Online

Code

C++ | Auto

```cpp
#include <iostream>
#include <vector>
 // Include the ListNode header
using namespace std;

class Solution {
public:
    ListNode* mergeTwoLists(ListNode* list1, ListNode* list2) {
        ListNode* dummy = new ListNode(0);  // Create a dummy node to start the merged list
        ListNode* current = dummy;  // Pointer to build the new list
```

Restored from local | Upgrade to Cloud Saving | Ln 1, Col 1

Testcase | Test Result

Case 1 | Case 2 | Case 3 | +

list1 =

[1,2,4]

list2 =

[1,3,4]

Source

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

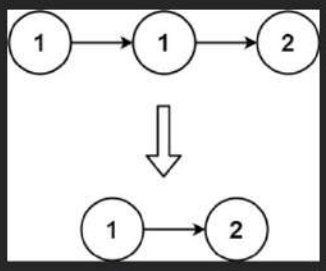
83. Remove Duplicates from Sorted List

Solved

Easy | Topics | Companies

Given the `head` of a sorted linked list, *delete all duplicates such that each element appears only once*. Return *the linked list* ***sorted*** *as well*.

Example 1:



Input: head = [1,1,2]
Output: [1,2]

Code

C++ | Auto

```cpp
#include <iostream>
using namespace std;

// Assume ListNode structure is already defined and precompiled.

class Solution {
public:
    ListNode* deleteDuplicates(ListNode* head) {
        ListNode* current = head;
        while (current != nullptr && current->next != nullptr) {
```

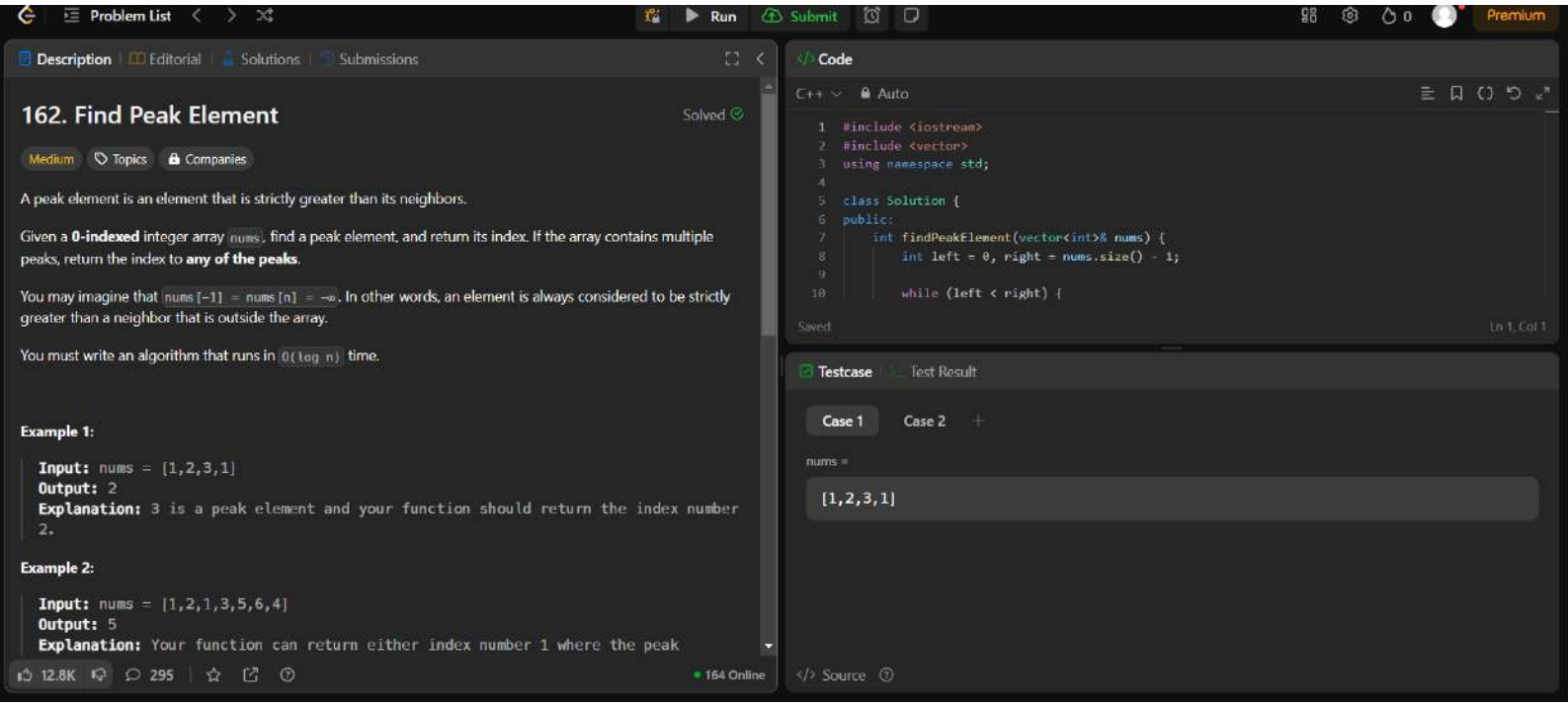
Restored from local | Upgrade to Cloud Saving | Ln 1, Col 1

Testcase | Test Result

Case 1 | Case 2 | +

head =

[1,1,2]


Problem List
Run
Submit
Premium
Description
Editorial
Solutions
Submissions
162. Find Peak Element
Solved
Medium
Topics
Companies
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 to any of the peaks.
You may imagine that nums[-1] = nums[n] = -∞. In other words, an element is always considered to be strictly greater than a neighbor that is outside the array.
You must write an algorithm that runs in O(log n) time.
Example 1:
Input: nums = [1,2,3,1]
Output: 2
Explanation: 3 is a peak element and your function should return the index number 2.
Example 2:
Input: nums = [1,2,1,3,5,6,4]
Output: 5
Explanation: Your function can return either index number 1 where the peak
12.8K
295
164 Online
Code
C++
Auto
#include <iostream>
#include <vector>
using namespace std;
class Solution {
public:
int findPeakElement(vector<int>& nums) {
int left = 0, right = nums.size() - 1;
while (left < right) {
Saved
Ln 1, Col 1
Testcase
Test Result
Case 1
Case 2
nums =
[1,2,3,1]
Source

Problem List

Run Submit

Premium

Description | Editorial | Solutions | Submissions

94. Binary Tree Inorder Traversal

Solved

Easy Topics Companies

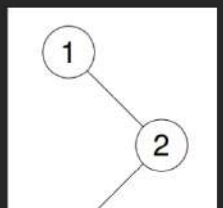
Given the `root` of a binary tree, return *the inorder traversal of its nodes' values.*

Example 1:

Input: `root = [1,null,2,3]`

Output: `[1,3,2]`

Explanation:



13.9K 188 151 Online

Code

C++ Auto

```
#include <iostream>
#include <vector>


using namespace std;

class Solution {
public:
    void inorderTraversalHelper(TreeNode* root, vector<int>& result) {
        if (root == nullptr) {
```

Restored from local Upgrade to Cloud Saving

Ln 1, Col

Testcase Test Result

Case 1 Case 2 Case 3 Case 4

root =

`[1,null,2,3]`

Source

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

51. N-Queens

Solved

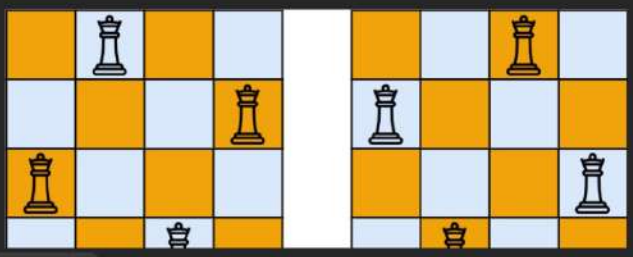
Hard | Topics | Companies

The **n-queens** puzzle is the problem of placing `n` queens on an `n x n` chessboard such that no two queens attack each other.

Given an integer `n`, return *all distinct solutions to the* ***n-queens puzzle***. You may return the answer in **any order**.

Each solution contains a distinct board configuration of the n-queens' placement, where `'Q'` and `'.'` both indicate a queen and an empty space, respectively.

Example 1:



12.9K | 118 | 160 Online

Code

C++ | Auto

```cpp
#include <vector>
#include <string>
using namespace std;

class Solution {
public:
    vector<vector<string>> solveNQueens(int n) {
        vector<vector<string>> solutions;
        vector<string> board(n, string(n, '.')); // Initialize an empty n x n board
        vector<int> leftRow(n, 0), upperDiag(2 * n - 1, 0), lowerDiag(2 * n - 1, 0);
```

Restored from local | Upgrade to Cloud Saving | Ln 1, Col 1

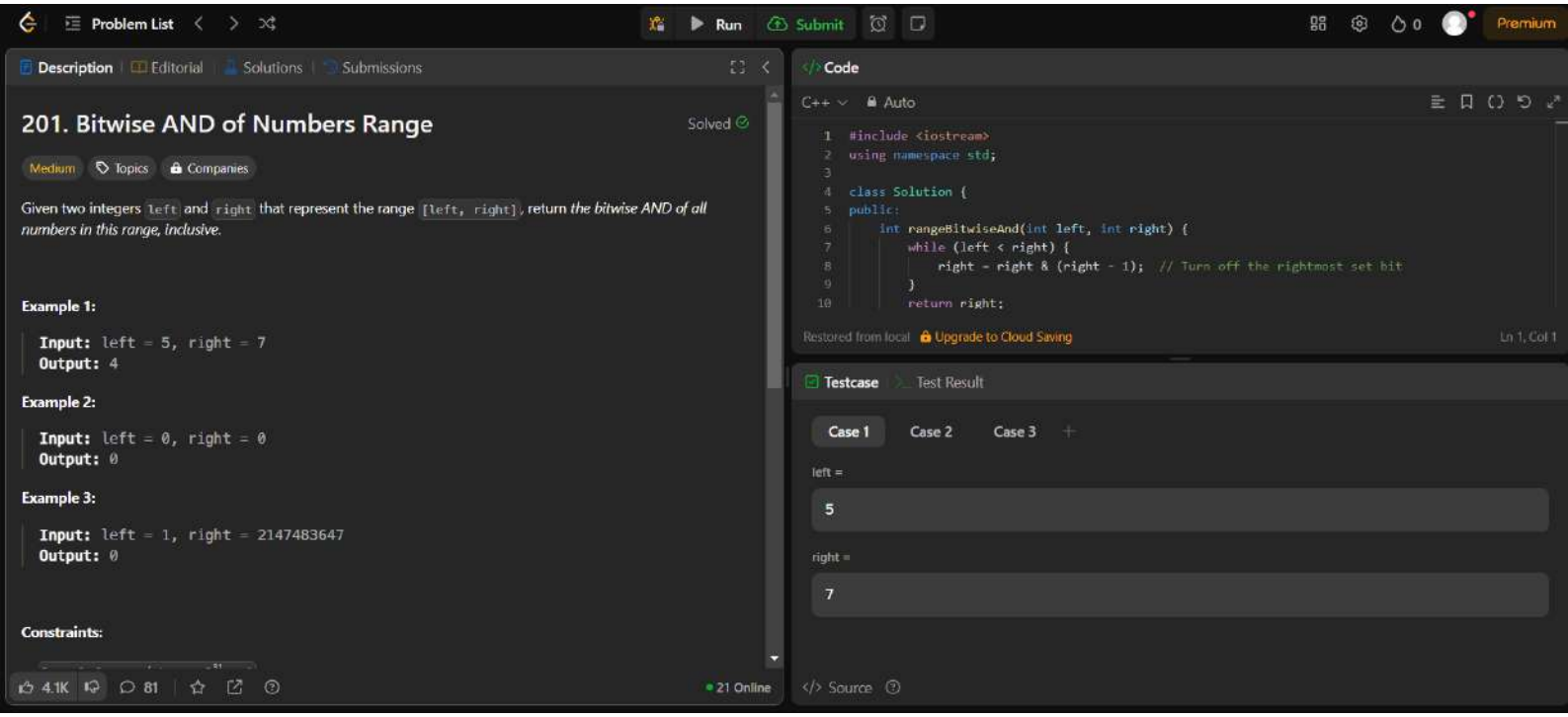
Testcase | Test Result

Case 1 | Case 2 | +

n =

4

Source


Problem List
Run
Submit
Premium
Description
Editorial
Solutions
Submissions
201. Bitwise AND of Numbers Range
Solved
Medium
Topics
Companies
Given two integers left and right that represent the range [left, right], return the bitwise AND of all numbers in this range, inclusive.
Example 1:
Input: left = 5, right = 7
Output: 4
Example 2:
Input: left = 0, right = 0
Output: 0
Example 3:
Input: left = 1, right = 2147483647
Output: 0
Constraints:
4.1K
81
21 Online
Code
C++
Auto
#include <iostream>
using namespace std;
class Solution {
public:
int rangeBitwiseAnd(int left, int right) {
while (left < right) {
right = right & (right - 1); // Turn off the rightmost set bit
}
return right;
Restored from local
Upgrade to Cloud Saving
Ln 1, Col 1
Testcase
Test Result
Case 1
Case 2
Case 3
left =
5
right =
7
Source