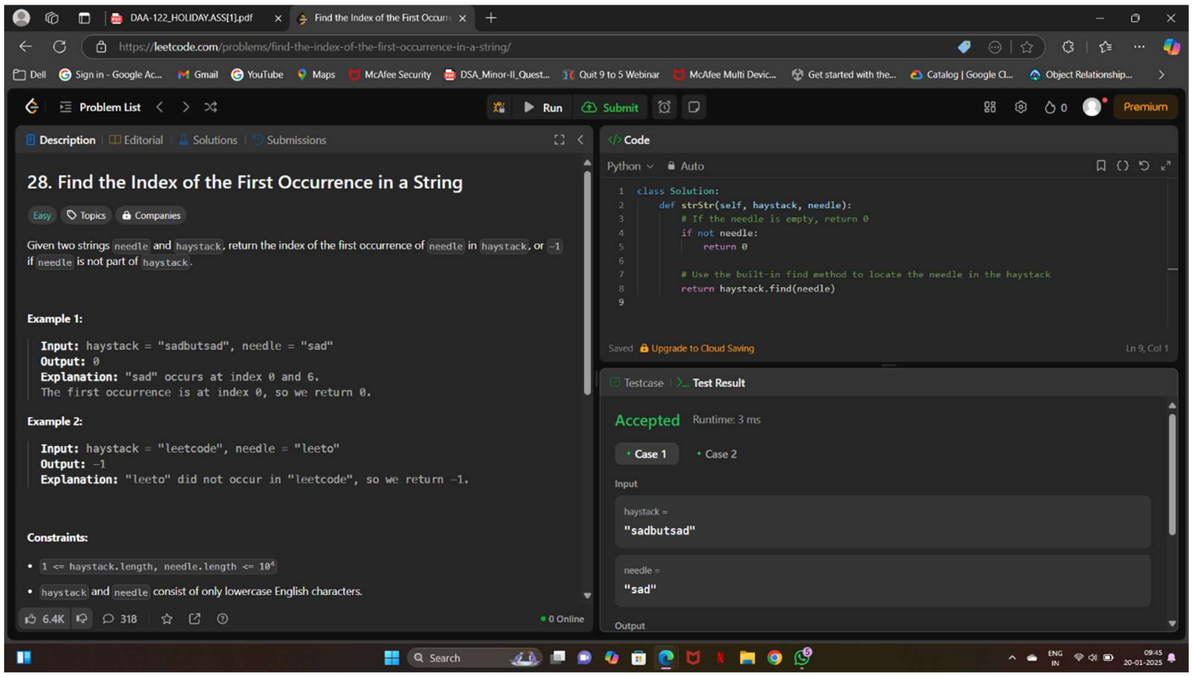
DAA HOLIDAY ASSIGNMENT

1) find-the-index-of-the-first-occurrence-in-a-string

https://leetcode.com/problems/find-the-index-of-the-first-occurrence-in-a-string/description/



https://leetcode.com/problems/find-the-index-of-the-first-occurrence-in-a-string/

28. Find the Index of the First Occurrence in a String

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 <= 10⁴`
- `haystack` and `needle` consist of only lowercase English characters.

Code — Python

```python
class Solution:
    def strStr(self, haystack, needle):
        # If the needle is empty, return 0
        if not needle:
            return 0

        # Use the built-in find method to locate the needle in the haystack
        return haystack.find(needle)

```

Saved · Upgrade to Cloud Saving · Ln 9, Col 1

Test Result

Accepted   Runtime: 3 ms

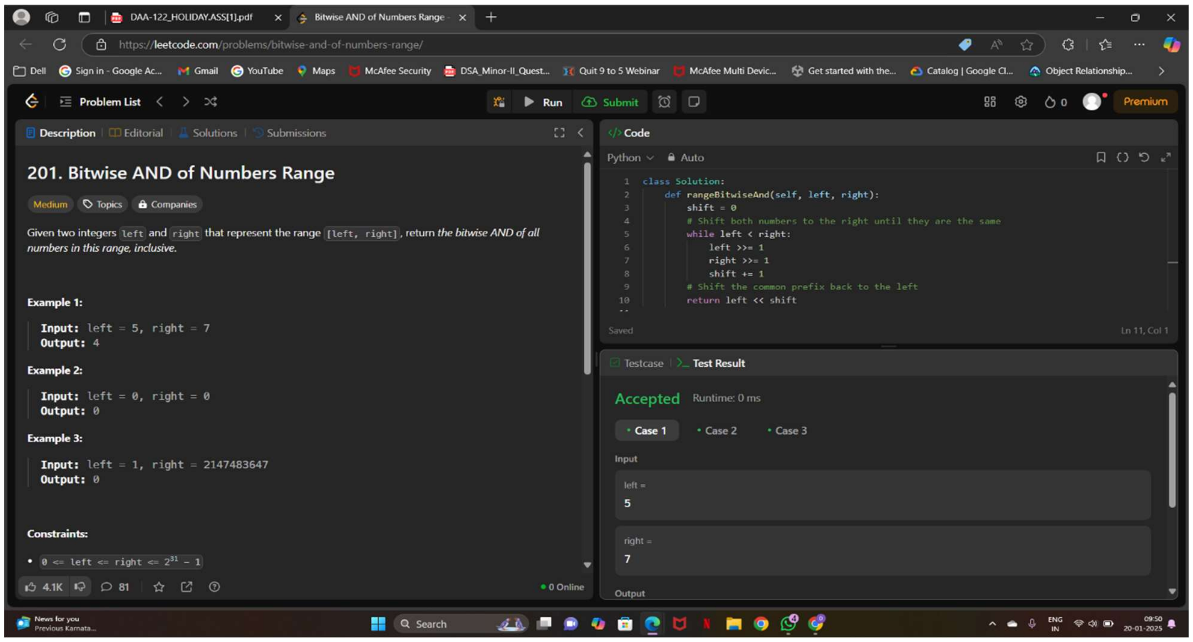
Case 1 · Case 2

Input

haystack =
```
"sadbutsad"
```
needle =
```
"sad"
```
Output

2) **Bitwise AND of numberrange**

https://leetcode.com/problems/bitwise-and-of-numbers-range/description/



https://leetcode.com/problems/bitwise-and-of-numbers-range/

201. Bitwise AND of Numbers Range

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:

- `0 <= left <= right <= 2³¹ - 1`

Code — Python

```python
class Solution:
    def rangeBitwiseAnd(self, left, right):
        shift = 0
        # Shift both numbers to the right until they are the same
        while left < right:
            left >>= 1
            right >>= 1
            shift += 1
        # Shift the common prefix back to the left
        return left << shift
```

Saved · Ln 11, Col 1

Test Result

Accepted   Runtime: 0 ms

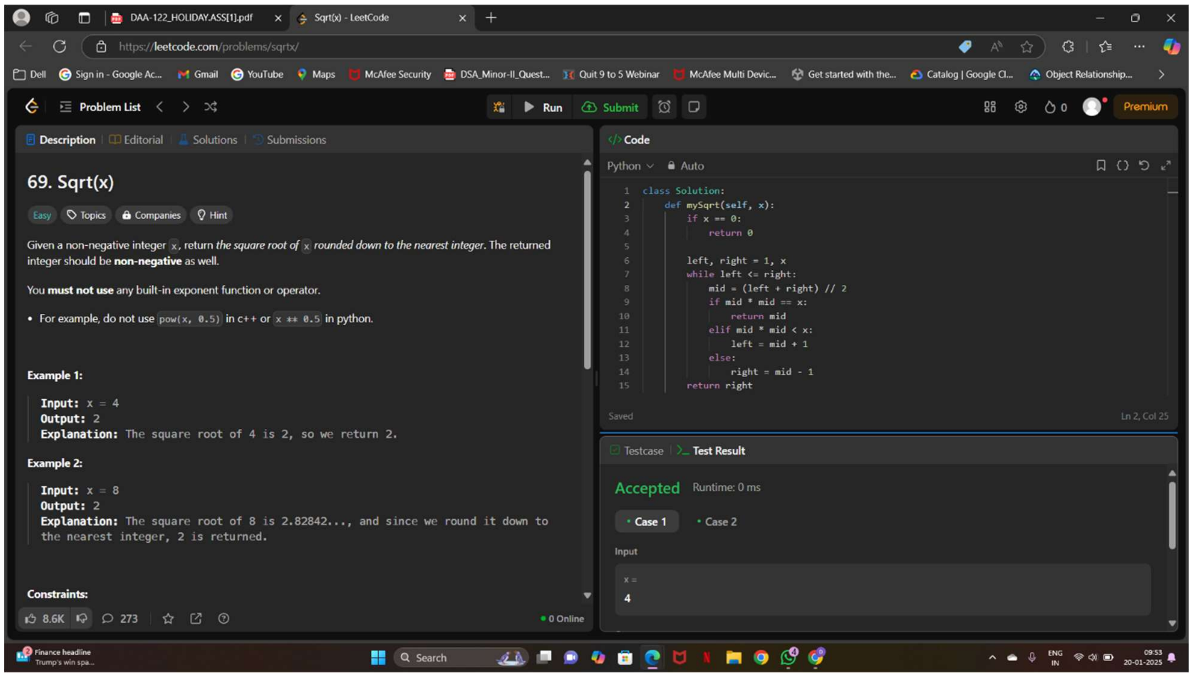
Case 1 · Case 2 · Case 3

Input

left =
```
5
```
right =
```
7
```
Output

3)SQRT(X)

https://leetcode.com/problems/sqrtx/submissions/1514635920/



69. Sqrt(x)

Easy

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.
```

Constraints:

```python
class Solution:
    def mySqrt(self, x):
        if x == 0:
            return 0

        left, right = 1, x
        while left <= right:
            mid = (left + right) // 2
            if mid * mid == x:
                return mid
            elif mid * mid < x:
                left = mid + 1
            else:
                right = mid - 1
        return right
```

Accepted Runtime: 0 ms

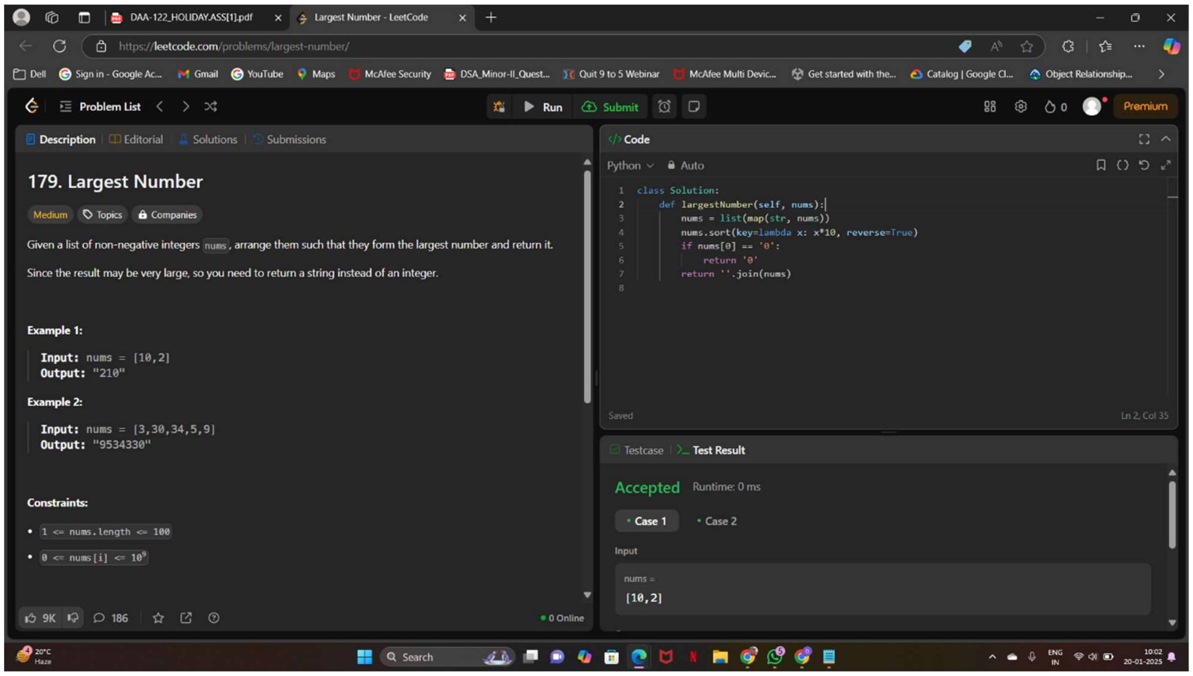
Case 1 Case 2

Input

x =
4

4)Largest Number

https://leetcode.com/problems/largest-number/description/



179. Largest Number

Medium

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}

```python
class Solution:
    def largestNumber(self, nums):
        nums = list(map(str, nums))
        nums.sort(key=lambda x: x*10, reverse=True)
        if nums[0] == '0':
            return '0'
        return ''.join(nums)
```

Accepted Runtime: 0 ms

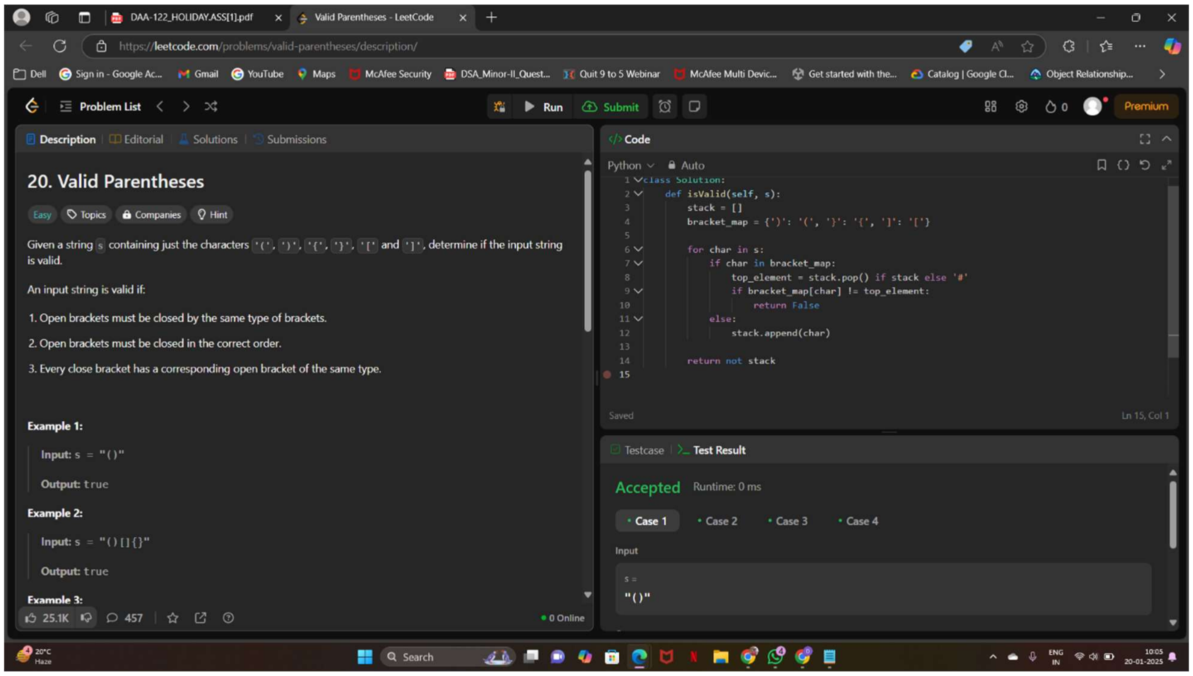
Case 1 Case 2

Input

nums =
[10,2]

5)Valid Parenthesis

https://leetcode.com/problems/valid-parentheses/description/



https://leetcode.com/problems/valid-parentheses/description/

20. Valid Parentheses

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

Example 3:

```python
class Solution:
    def isValid(self, s):
        stack = []
        bracket_map = {')': '(', '}': '{', ']': '['}

        for char in s:
            if char in bracket_map:
                top_element = stack.pop() if stack else '#'
                if bracket_map[char] != top_element:
                    return False
            else:
                stack.append(char)

        return not stack

```

Saved · Ln 15, Col 1

Accepted Runtime: 0 ms

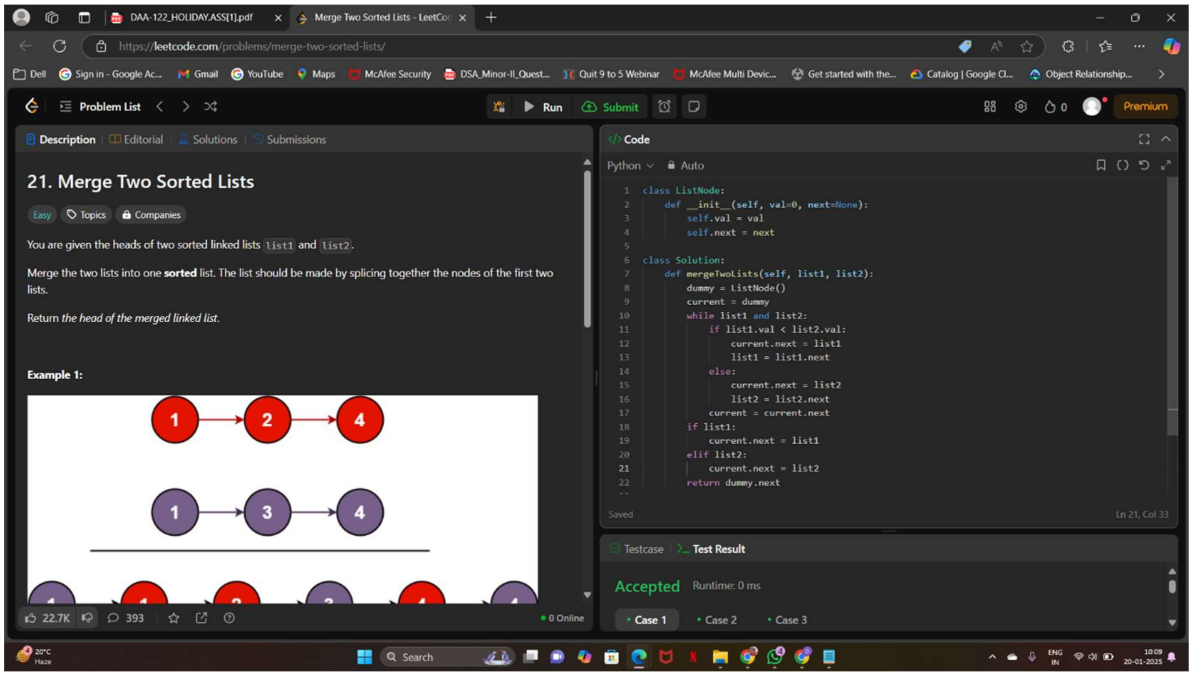
Case 1 · Case 2 · Case 3 · Case 4

Input

s =
"()"

6)Merge Two Sorted Lists

https://leetcode.com/problems/merge-two-sorted-lists/description/



https://leetcode.com/problems/merge-two-sorted-lists/

21. Merge Two Sorted Lists

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:

```python
class ListNode:
    def __init__(self, val=0, next=None):
        self.val = val
        self.next = next

class Solution:
    def mergeTwoLists(self, list1, list2):
        dummy = ListNode()
        current = dummy
        while list1 and list2:
            if list1.val < list2.val:
                current.next = list1
                list1 = list1.next
            else:
                current.next = list2
                list2 = list2.next
            current = current.next
        if list1:
            current.next = list1
        elif list2:
            current.next = list2
        return dummy.next
```

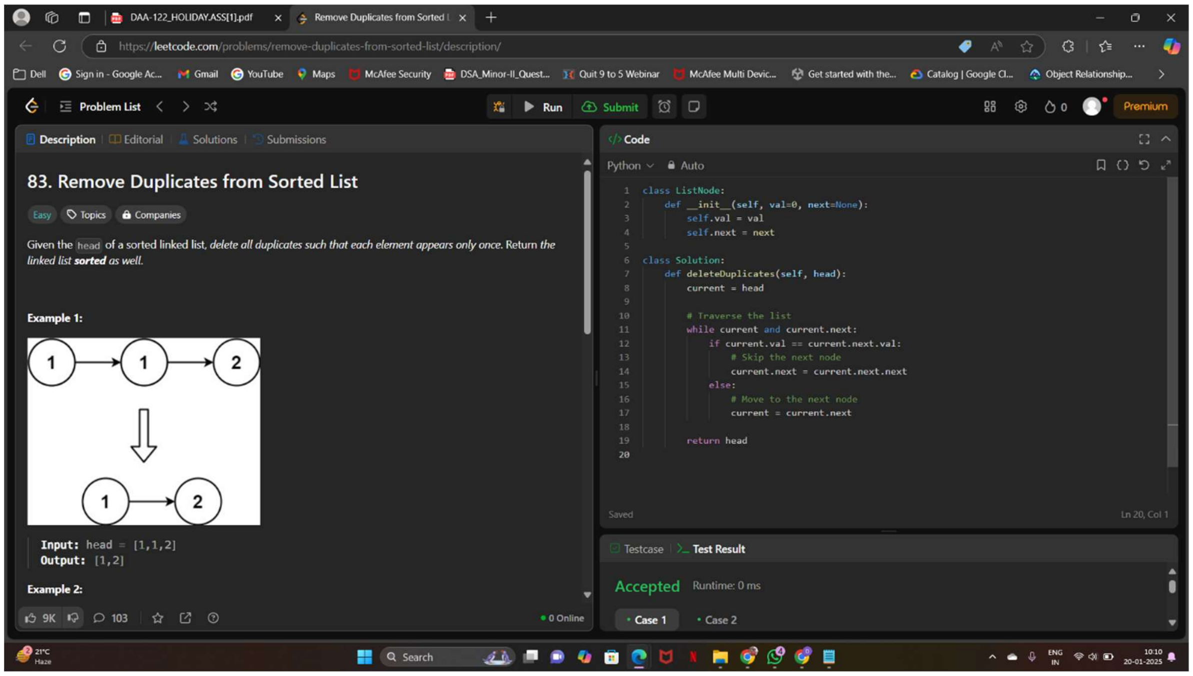
Saved · Ln 21, Col 33

Accepted Runtime: 0 ms

Case 1 · Case 2 · Case 3

7)Remove Duplicates from sorted list

https://leetcode.com/problems/remove-duplicates-from-sorted-list/description/



83. Remove Duplicates from Sorted List

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]
```

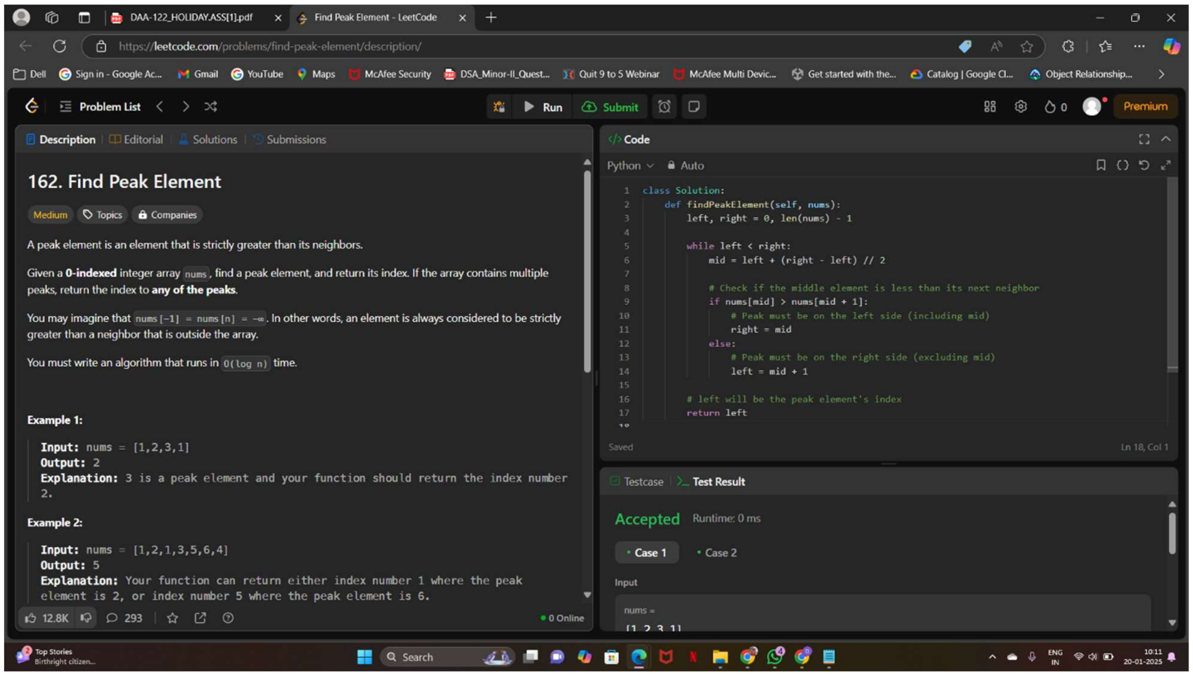
Example 2:

```python
class ListNode:
    def __init__(self, val=0, next=None):
        self.val = val
        self.next = next

class Solution:
    def deleteDuplicates(self, head):
        current = head

        # Traverse the list
        while current and current.next:
            if current.val == current.next.val:
                # Skip the next node
                current.next = current.next.next
            else:
                # Move to the next node
                current = current.next

        return head
```

Accepted Runtime: 0 ms

8)Find peak Element

https://leetcode.com/problems/find-peak-element/description/



162. Find Peak Element

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
element is 2, or index number 5 where the peak element is 6.
```

```python
class Solution:
    def findPeakElement(self, nums):
        left, right = 0, len(nums) - 1

        while left < right:
            mid = left + (right - left) // 2

            # Check if the middle element is less than its next neighbor
            if nums[mid] > nums[mid + 1]:
                # Peak must be on the left side (including mid)
                right = mid
            else:
                # Peak must be on the right side (excluding mid)
                left = mid + 1

        # left will be the peak element's index
        return left
```

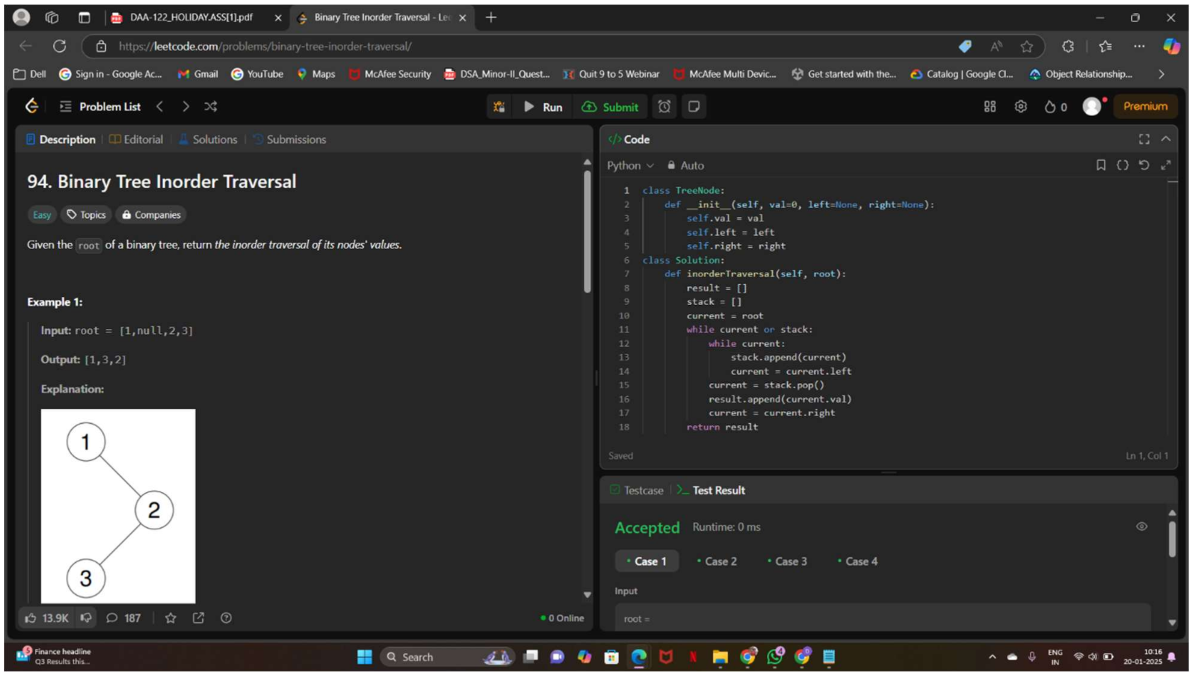
Accepted Runtime: 0 ms

Input

nums =
[1,2,3,1]

9)Binary Tree Inorder Traversal

https://leetcode.com/problems/binary-tree-inorder-traversal/submissions/1514639992/



94. Binary Tree Inorder Traversal

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:

```python
class TreeNode:
    def __init__(self, val=0, left=None, right=None):
        self.val = val
        self.left = left
        self.right = right
class Solution:
    def inorderTraversal(self, root):
        result = []
        stack = []
        current = root
        while current or stack:
            while current:
                stack.append(current)
                current = current.left
            current = stack.pop()
            result.append(current.val)
            current = current.right
        return result
```

Accepted Runtime: 0 ms

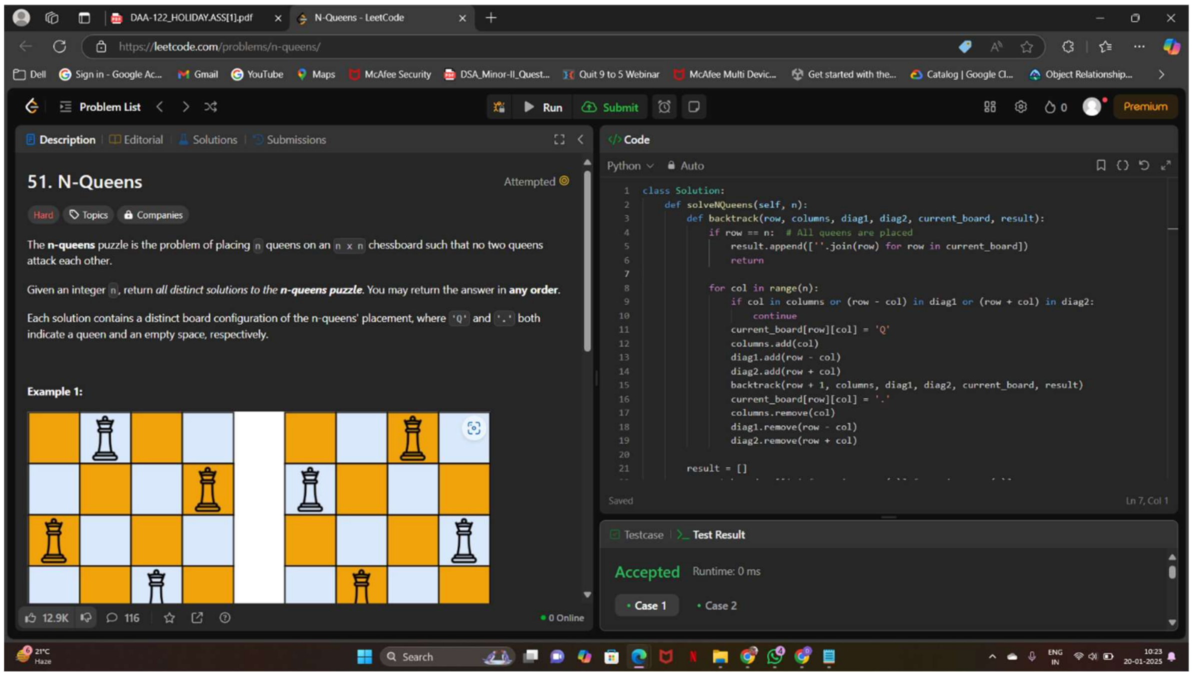
Case 1 · Case 2 · Case 3 · Case 4

Input

root =

10)N-Queens

https://leetcode.com/problems/n-queens/description/



51. N-Queens

Attempted

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:

```python
class Solution:
    def solveNQueens(self, n):
        def backtrack(row, columns, diag1, diag2, current_board, result):
            if row == n:  # All queens are placed
                result.append([''.join(row) for row in current_board])
                return

            for col in range(n):
                if col in columns or (row - col) in diag1 or (row + col) in diag2:
                    continue
                current_board[row][col] = 'Q'
                columns.add(col)
                diag1.add(row - col)
                diag2.add(row + col)
                backtrack(row + 1, columns, diag1, diag2, current_board, result)
                current_board[row][col] = '.'
                columns.remove(col)
                diag1.remove(row - col)
                diag2.remove(row + col)

        result = []
```

Accepted Runtime: 0 ms

Case 1 · Case 2

Submitted By:

D. Sai Kartheek

2211CS020119

AIML-BETA