## CHANDIGARH UNIVERSITY

## UNIVERSITY INSTITUTE OF ENGINEERING

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

****

|  |  |
| --- | --- |
| **Submitted By: Submitted To:**  Vivek Kumar(21BCS8129) Anum Malike(E14644) | |
| **Subject Name** | Mobile Application Development |
| **Subject Code** | 20CSP-351 |
| **Branch** | Computer Science and Engineering |
| **Semester** | 6th |

**Experiment No. - 1**

**Student Name: Vivek Kumar UID: 21BCS8129**

**Branch: BE-CSE(LEET) Section/Group: 20BCS-ST-801/B**

**Semester: 6th Date of Performance: 14/02/2023**

**Subject Name: Competitive coding - II Subject Code: 20CSP-351**

1. **Aim/Overview of the practical:**

**Jump Game II**

You are given a 0-indexed array of integers nums of length n. You are initially positioned at nums[0].

Each element nums[i] represents the maximum length of a forward jump from index i. In other words, if you are at nums[i], you can jump to any nums[i + j] where:

* 0 <= j <= nums[i] and
* i + j < n

Return the minimum number of jumps to reach nums[n - 1]. The test cases are generated such that you can reach nums[n - 1].

https://leetcode.com/problems/jump-game-ii/

1. **Apparatus / Simulator Used:**

* Windows 7 or above
* Google Chrome

1. **Objective:**
   * To understand the concept of Array and Jump Concept
   * To implement the concept of Array Implementation.

**4. Code:**

class Solution {

public int jump(int[] nums) {

int answer = 0, n = nums.length;

int curEnd = 0, curFar = 0;

for (int i = 0; i < n - 1; ++i) {

curFar = Math.max(curFar, i + nums[i]);

if (i == curEnd) {

answer++;

curEnd = curFar;

}

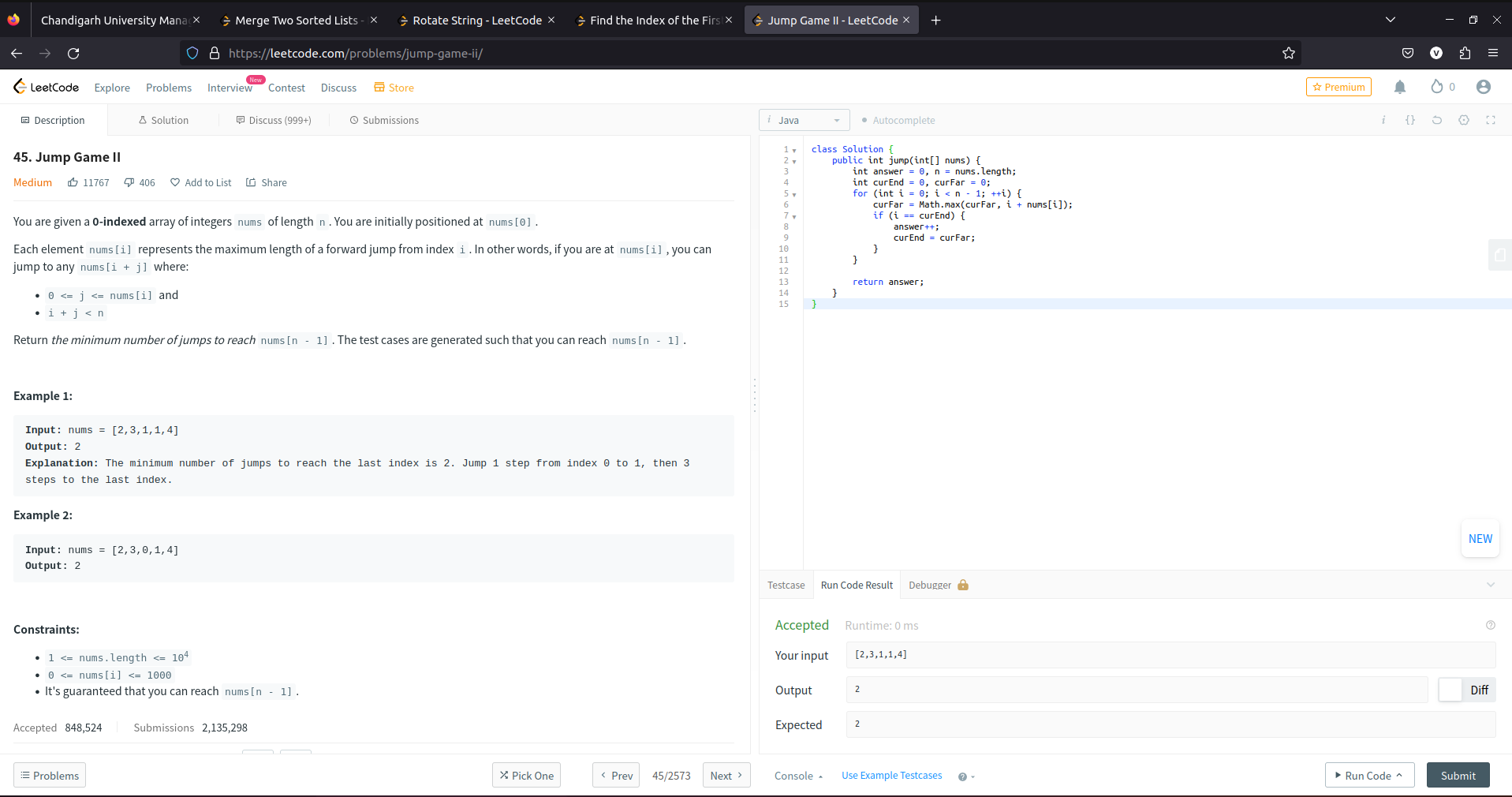
}

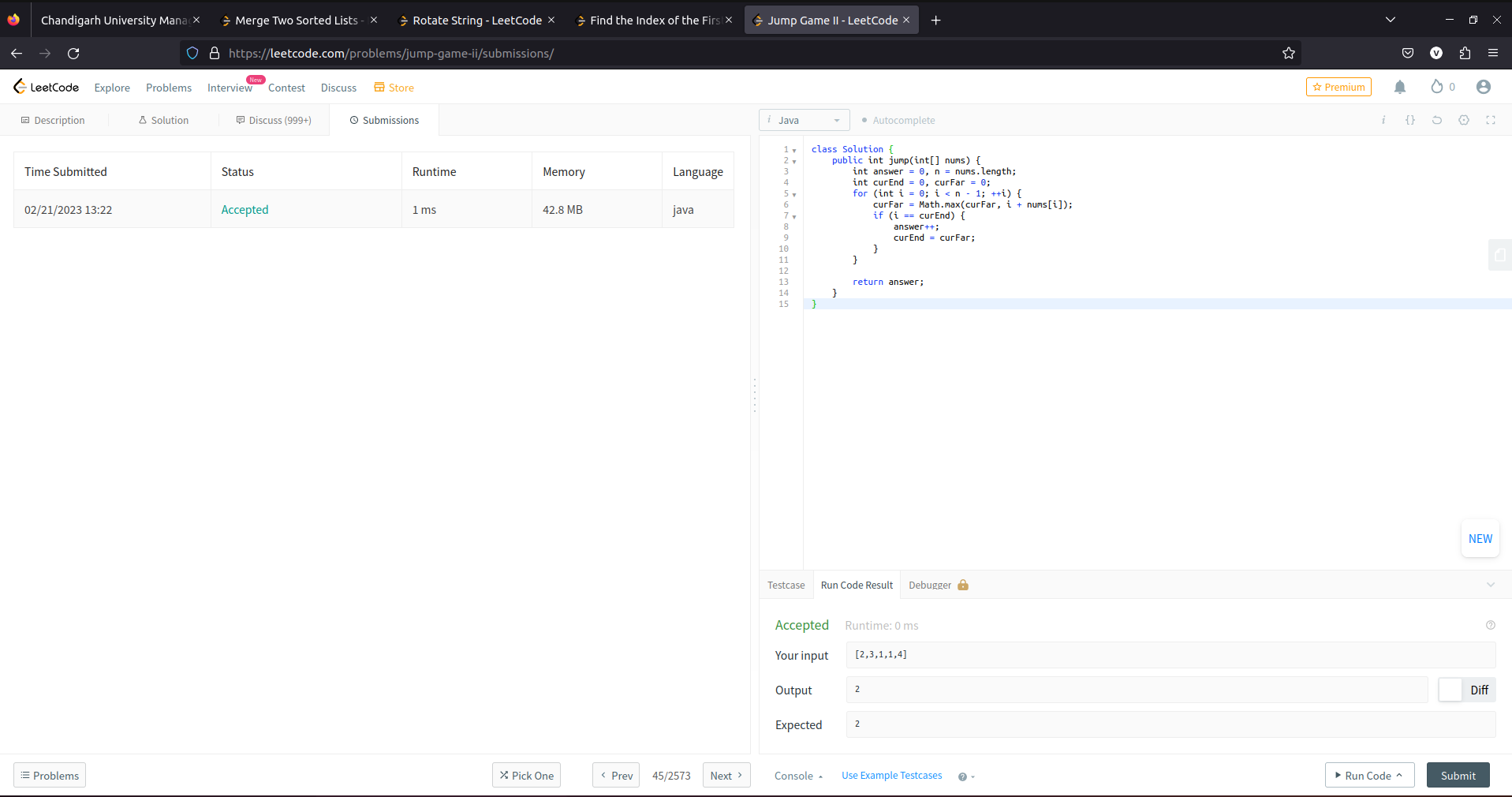
return answer;

}

}

**5. Result/Output/Writing Summary:**





1. **Aim/Overview of the practical:**

**Merge Two Sorted List**

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

Merge the two lists in a 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.

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

1. **Apparatus / Simulator Used:**

* Windows 7 or above
* Google Chrome

1. **Objective:**
   * To understand the concept of List and Node
   * To implement the concept of Sorting and Merge.
2. **Code:**

class Solution {

public ListNode mergeTwoLists(ListNode l1, ListNode l2) {

ListNode result = new ListNode();

ListNode head = result;

while(l1!=null && l2!=null){

if(l1.val<l2.val){

result.next = l1;

l1 = l1.next;

}else{

result.next = l2;

l2 = l2.next;

}

result = result.next;

}

if(l1!=null){

result.next=l1;

}else{

result.next=l2;

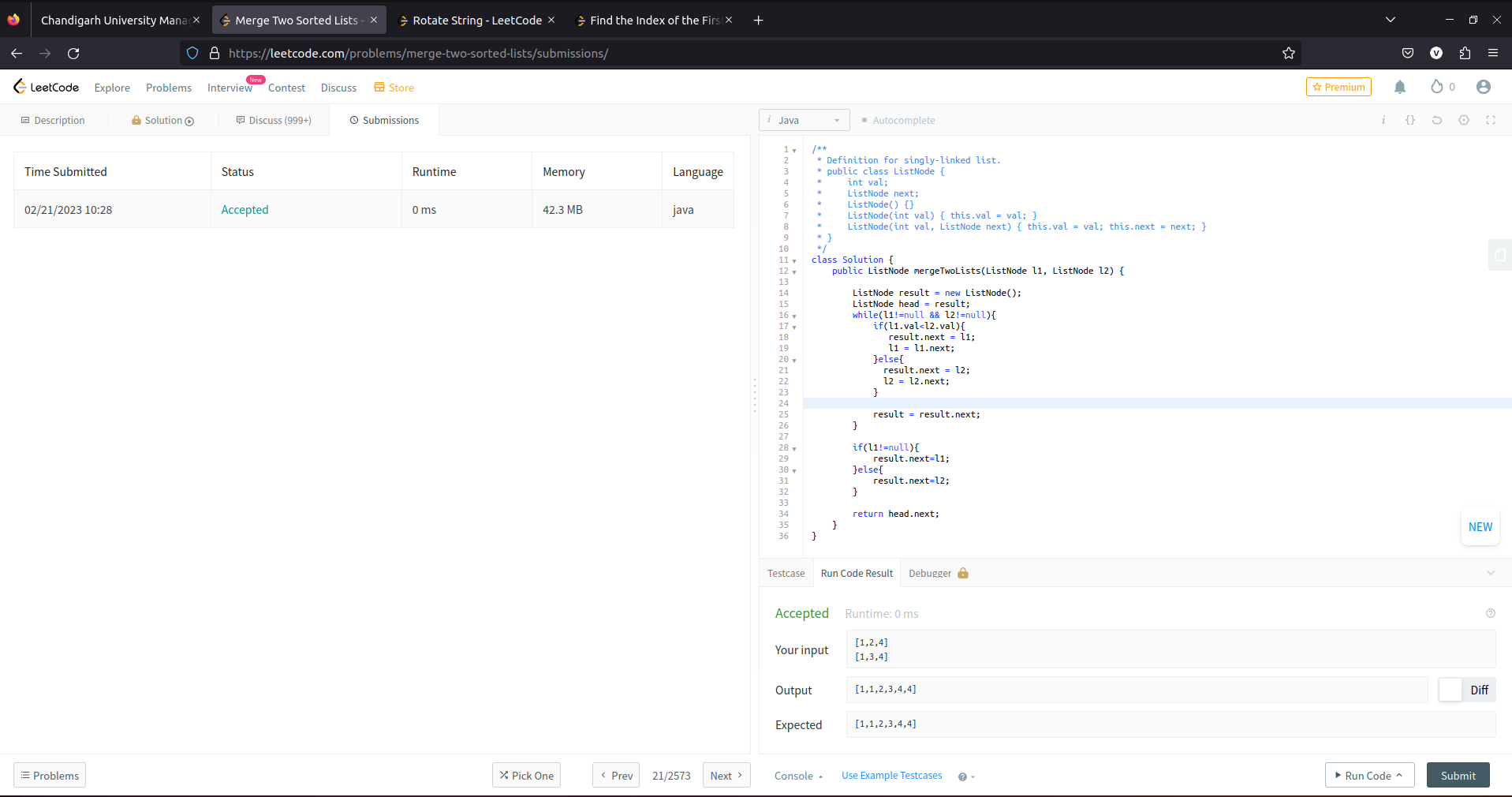
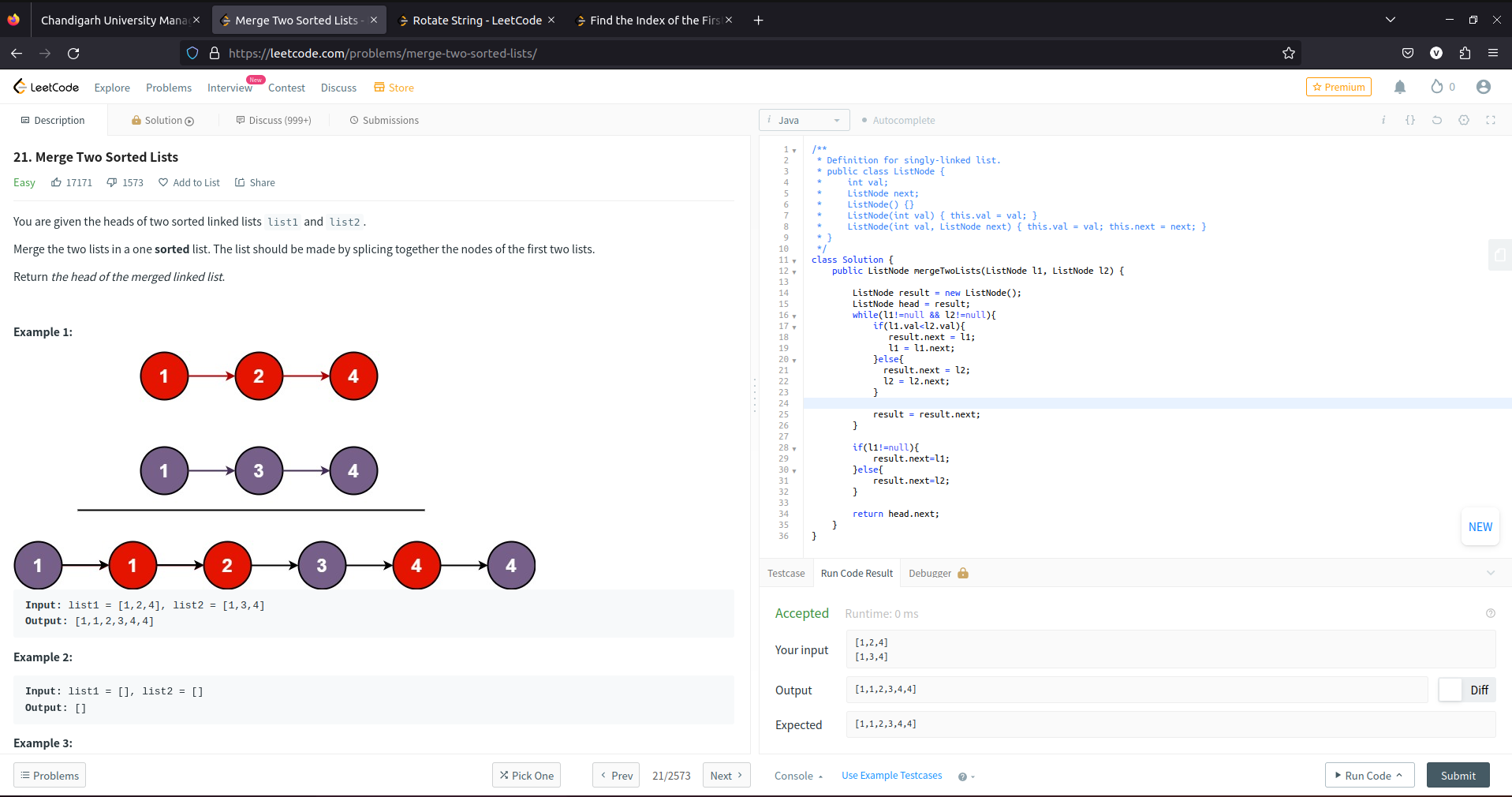
}

return head.next;

}

}

1. **Result/Output/Writing Summary:**

****

**Learning outcomes (What I have learnt):**

* + Learned the concept of LinkedList.
  + Learnt about Array in Merging And Sorting.

**Evaluation Grid (To be created per the faculty's SOP and Assessment guidelines):**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. | Worksheet completion including writing learning objectives/Outcomes.  (To be submitted at the end of the day). |  |  |
| 2. | Post-Lab Quiz Result. |  |  |
| 3. | Student Engagement in  Simulation/Demonstration/Performance and Controls/Pre-Lab Questions. |  |  |
|  | Signature of Faculty (with Date): | Total Marks Obtained: |  |