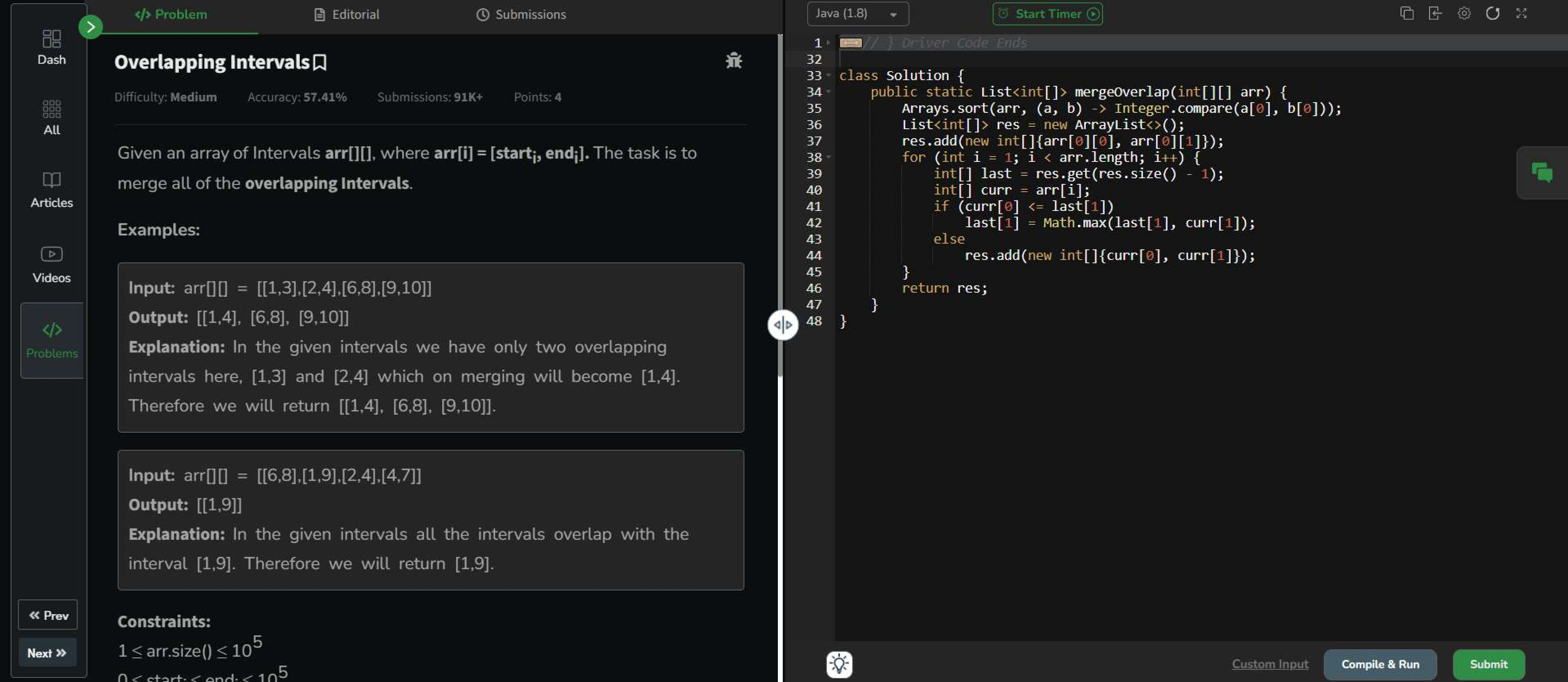

Overlapping Intervals

Difficulty: **Medium** Accuracy: **57.41%** Submissions: **91K+** Points: **4**

Given an array of Intervals **arr[][]**, where **arr[i] = [start$_i$, end$_i$]**. The task is to merge all of the **overlapping Intervals**.

Examples:

Input: arr[][] = [[1,3],[2,4],[6,8],[9,10]]
Output: [[1,4], [6,8], [9,10]]
Explanation: In the given intervals we have only two overlapping intervals here, [1,3] and [2,4] which on merging will become [1,4]. Therefore we will return [[1,4], [6,8], [9,10]].

Input: arr[][] = [[6,8],[1,9],[2,4],[4,7]]
Output: [[1,9]]
Explanation: In the given intervals all the intervals overlap with the interval [1,9]. Therefore we will return [1,9].

Constraints:
$1 \le arr.size() \le 10^5$
$0 \le start_i \le end_i \le 10^5$

```java
// } Driver Code Ends

class Solution {
    public static List<int[]> mergeOverlap(int[][] arr) {
        Arrays.sort(arr, (a, b) -> Integer.compare(a[0], b[0]));
        List<int[]> res = new ArrayList<>();
        res.add(new int[]{arr[0][0], arr[0][1]});
        for (int i = 1; i < arr.length; i++) {
            int[] last = res.get(res.size() - 1);
            int[] curr = arr[i];
            if (curr[0] <= last[1])
                last[1] = Math.max(last[1], curr[1]);
            else
                res.add(new int[]{curr[0], curr[1]});
        }
        return res;
    }
}
```