Merge Intervals

Try to solve the Merge Intervals problem.

We'll cover the following

^

- Statement
- Examples
- Understand the problem
- Figure it out!
- Try it yourself

Statement

We are given an array of **closed intervals**, **intervals**, where each interval has a start time and an end time. The input array is sorted with respect to the start times of each interval. For example, **intervals** = [1,4], [3,6], [7,9] is sorted in terms of start times 1, 3, and 7.

Your task is to merge the <u>overlapping intervals</u> and return a new output array consisting of only the non-overlapping intervals.

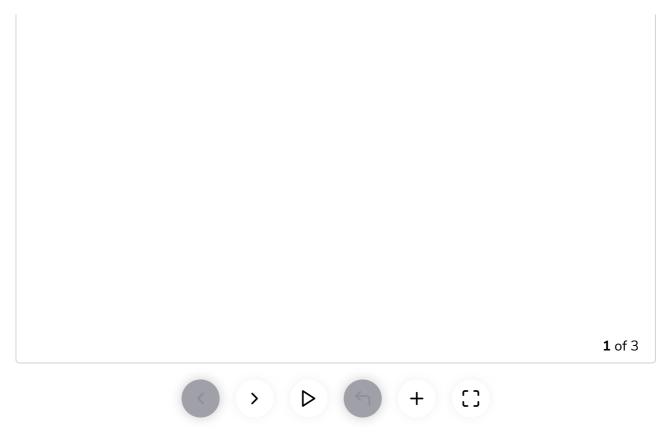
Constraints:

- $1 \leq { t intervals.length} \leq 10^4$
- intervals[i].length =2
- $0 \leq \text{start time} \leq \text{end time} \leq 10^4$

Examples

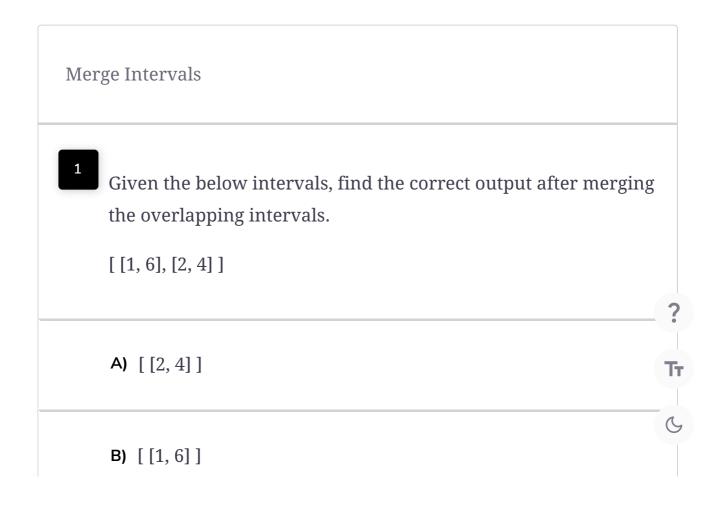
Тт





Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:



C) [[1, 6], [2, 4]]

D) [[2, 4], [1, 6]]

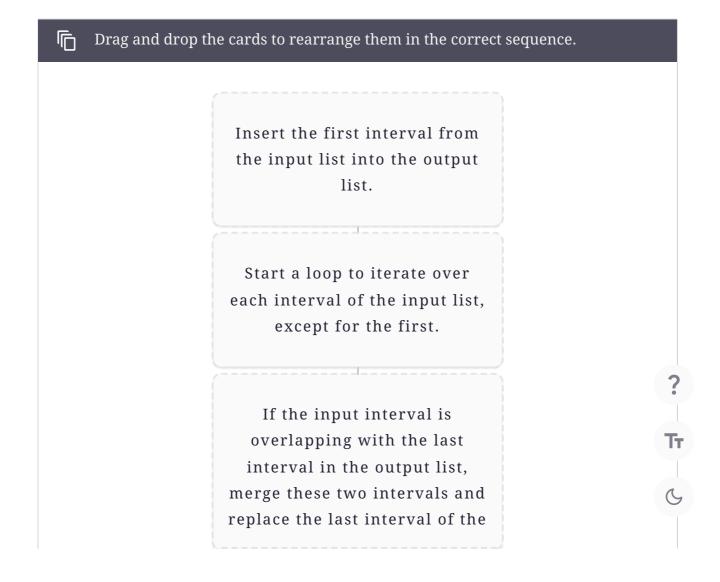
Submit Answer

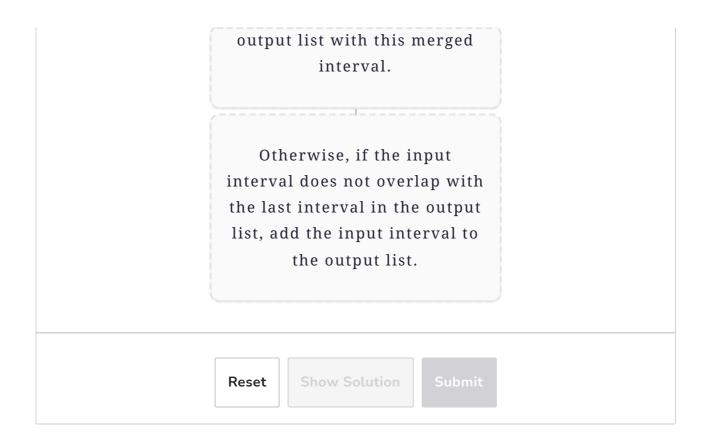
Question 1 of 4
0 attempted

Reset Quiz C

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.









Implement your solution in MergeInterval.java in the following coding playground. We have provided the definition of the Interval class in the other file. You may add functionality to this class if you wish.





