

Merge Overlapping Intervals

56. Merge Intervals

Solved

Medium Topics Companies

Given an array of intervals where $\text{intervals}[i] = [\text{start}_i, \text{end}_i]$, merge all overlapping intervals, and return an array of the non-overlapping intervals that cover all the intervals in the input.

Example 1:

Input: $\text{intervals} = [[1,3], [2,6], [8,10], [15,18]]$

Output: $[[1,6], [8,10], [15,18]]$

Explanation: Since intervals $[1,3]$ and $[2,6]$ overlap, merge them into $[1,6]$.

Example 2:

Input: $\text{intervals} = [[1,4], [4,5]]$

Output: $[[1,5]]$

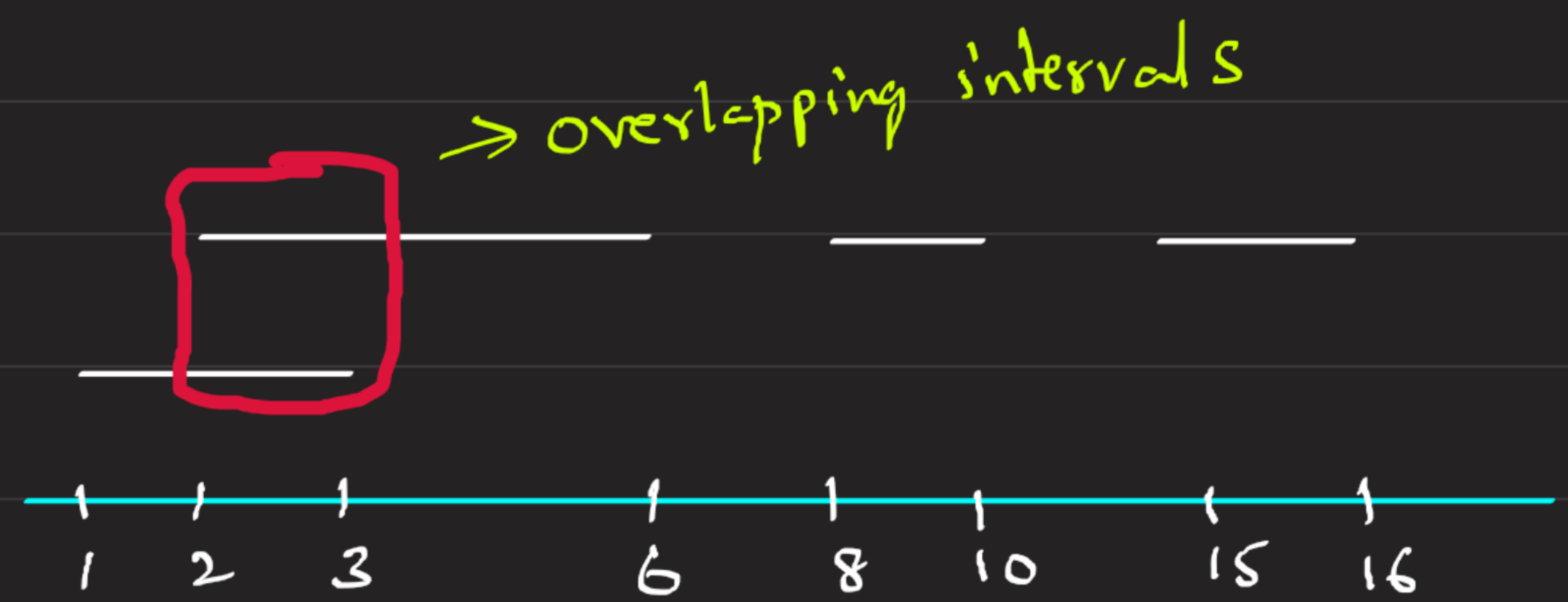
Explanation: Intervals $[1,4]$ and $[4,5]$ are considered overlapping.

Constraints:

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

First sort the array intervals

Ex:- $[[1,3], [2,6], [8,10], [15,18]]$



merged interval
↓

1 2 3 6 8 10 15 16

∴ Ans $\Rightarrow [[1,6], [8,10], [15,16]]$

* We have to make sure that, in final ans, there won't be any overlapping intervals

Approach

* Iterate from $0 \rightarrow n$

↳

as initially, our ans list is empty, we add first element into it

Observations

if $(\text{currentInterval}[i][0] < \text{lastInsertedArray}[1])$

$\text{lastInsertedArray}[1] = \max(\text{currentInterval}[i][1], \text{lastInsertedArray}[1])$

Case-I

If the current interval can be merged with the last inserted interval of the answer list

comparator 1
Ex: $[0, 2], [1, 4], [8, 10]$
ans list $[0, 2]$ → last inserted interval Array
1 is in range of the list,
So update interval end as $\max(4, 2)$

Update

maximum (Current Interval's End, last inserted interval End)

Case-II

If the current interval cannot be merged with last inserted interval of answer list



We insert the current interval in the answer list and update last inserted interval as current interval

if (currentInterval[i][0] > lastInsertedInterval[i]) → ignore

For your Benefit

To sort 2D arrays

① Arrays.sort(arr, (a, b) → Integer.compare(a[0], b[0]))
Any 2 random variables
this says based on 0th index sorting
array name
if you put a first → increasing order

Now, if i want a 2D array to sort descending order

Arrays.sort(arr, (a, b) → Integer.compare(b[0], a[0]))

