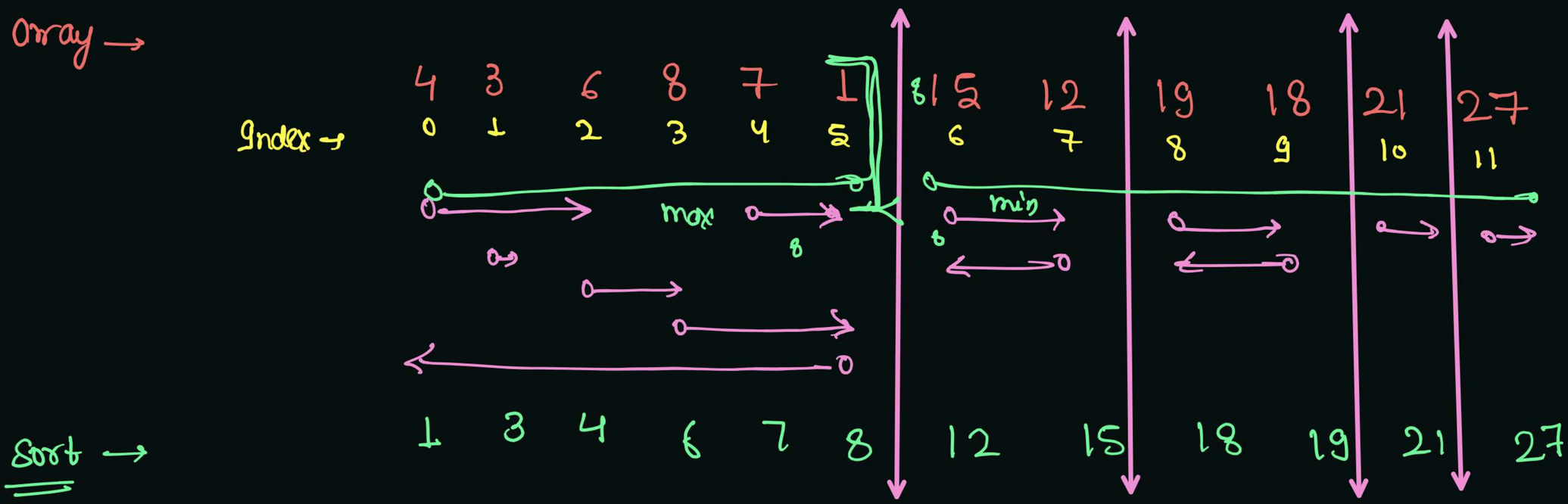
Partition Array Into Disjoint Intervals

Thursday, 16 September 2021 8:16 PM

Problem is similar to max. no. of chunks to make array Sorted - 2 ⟹



left partition's length: index + 1 = 5 + 1 = 6

First time left max < right min → answer is found

How to solve in O(n) space → left max and right min

array → 8 2 10 6 12 1 16 15 22 12 40

0 1 2 3 4 5 6 7 8 9 10

max. in partition

leftmax = ~~8~~ ~~10~~ 12

maxInRun = ~~8~~ ~~10~~ ~~12~~ ~~16~~ ~~22~~ 40

ans = ~~0~~ ~~1~~ ~~3~~ 5

Max. in Running window

// to know greater in upcoming iterations

if(arr[i] > maxInRun)

maxInRun = arr[i];

// To know impact till now

if(arr[i] < leftmax)

ans = i

leftmax = maxInRun

Result = ans + 1 = 5 + 1 = 6 length of left partition

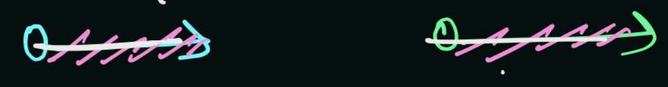
Merge Intervals

Thursday, 16 September 2021 8:16 PM

[1,3],[2,6],[8,10],[15,18]

Steps for Solution

1. Sort the array on the basis of starting time.
2. Travel and merge intervals in list.
3. Convert list into array.



2 platform

[1 - 6]

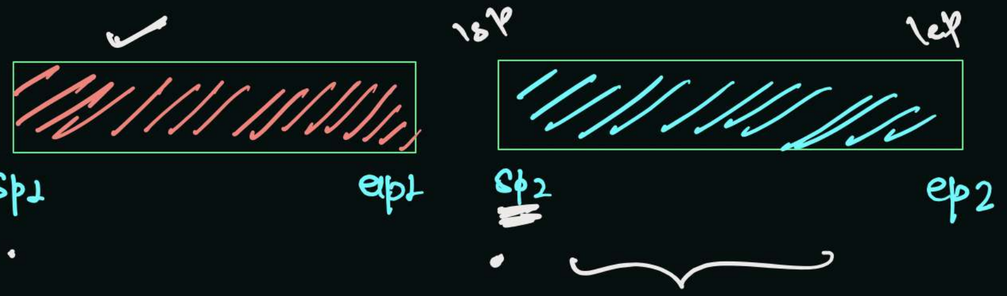
[8 - 10]

[15 - 18]

2

If we sort array on basis of starting_point

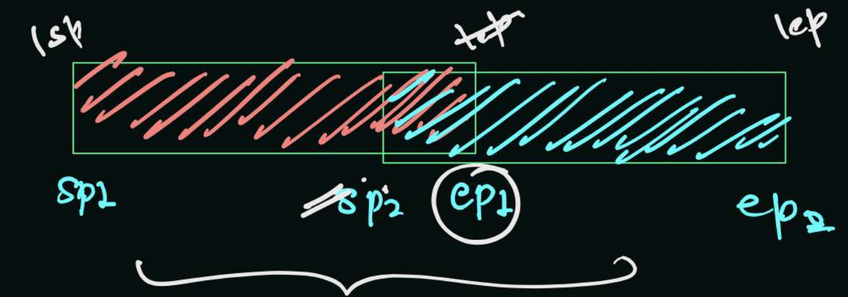
Case-I



[↳ ep1 < sp2 ⟶ No overlapping

[sp1 – ep1] – interval in Result

Case-II



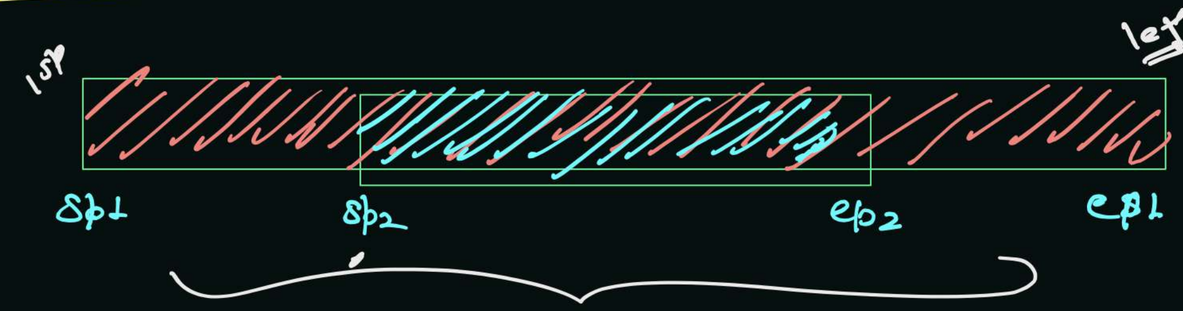
ep1 > sp2 ⟶ Overlapping

lep < ep.

⟶ ep1 < ep2 ⟶ partial overlapping

update ending point in merging interval

Case-III



ep1 > sp2 ⟶ overlapping.

⟶ ep1 > ep2 ⟶ Fully overlapping

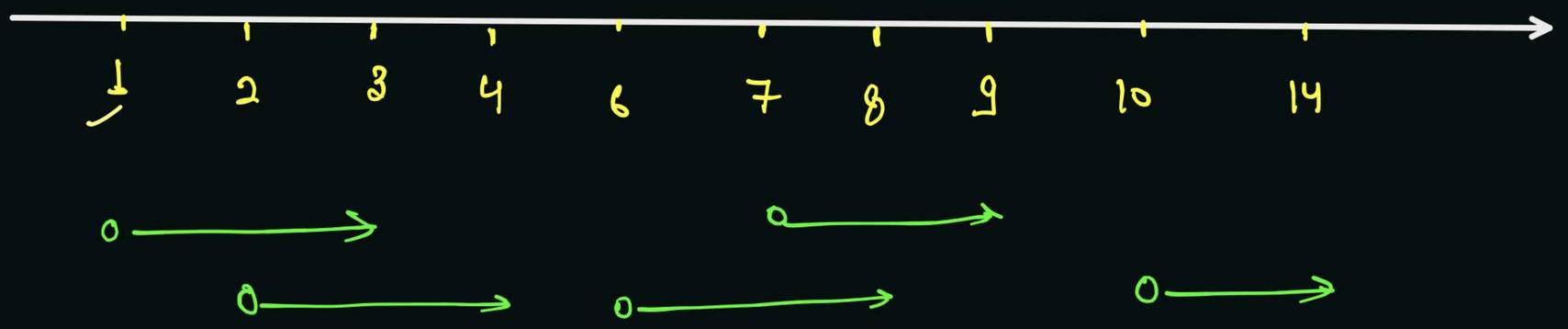
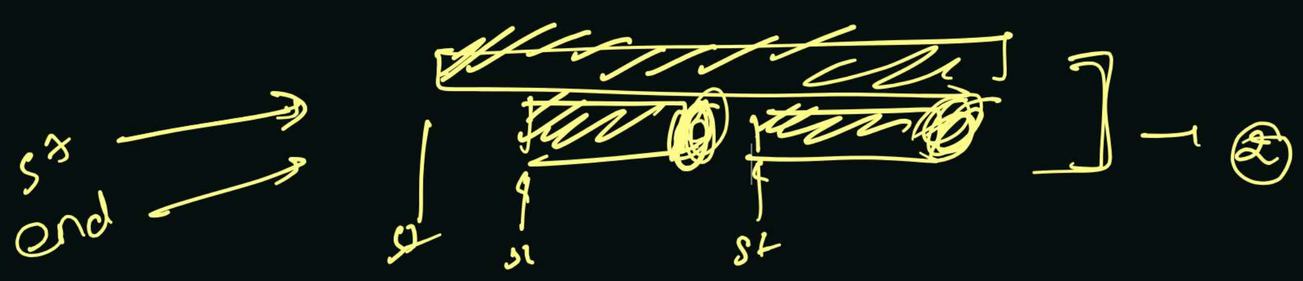
↳ continue to analyze next starting point.

Meeting Rooms 2

Thursday, 16 September 2021 8:16 PM

[[1,3],[2,4],[6,8],[10,14],[7,9]]

Max no. of meetings at a particular time??

⇓

max. no. of trains at particular time.