

253. Meeting Rooms II

Medium 2451 41 Add to List Share

Given an array of meeting time intervals consisting of start and end times $[[s_1,e_1],[s_2,e_2],\dots]$ ($s_i < e_i$), find the minimum number of conference rooms required.

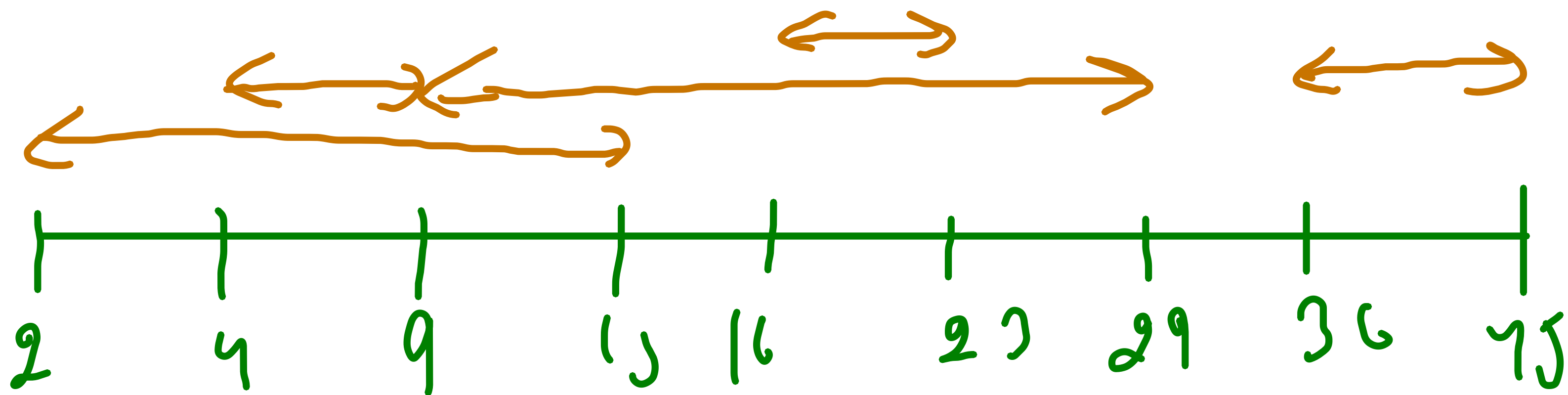
Example 1:

Input: $[[0, 30],[5, 10],[15, 20]]$
Output: 2

Example 2:

Input: $[[7,10],[2,4]]$
Output: 1

Input $[[2,15],[36,45],[9,29],[16,23],[4,9]]$
Output 3



→ jb bhi meeting start hogi +1 or jb khatam hogi toh -1 esa krke time maximum no. roomy jo honge vo answer hai.
→ or iski jgha treemap ka bhi use krsktte hai.

- 1. Load all intervals to the `TreeMap`, where keys are intervals' start/end boundaries, and values accumulate the changes at that point in time.
- 2. Traverse the `TreeMap` (in other words, sweep the timeline). If a new interval starts, increase the counter (`val` value) by 1, and the counter decreases by 1, if an interval has finished.
- 3. Calculate the number of the active ongoing intervals.

```
//using sorting.=====
int minMeetingRooms_01(vector<vector<int>> &intervals)
{
    if (intervals.size() <= 1)
        return intervals.size();

    int n = intervals.size();
    vector<int> start;
    vector<int> end;
    for (auto &ar : intervals)
    {
        start.push_back(ar[0]);
        end.push_back(ar[1]);
    }

    sort(start.begin(), start.end());
    sort(end.begin(), end.end());

    int count = 0, i = 0, j = 0, ans = 0;
    while (i < n && j < n)
    {
        if (start[i] < end[j])
            count++, i++;
        else
            count--, j++;
        ans = max(ans, count);
    }
    return ans;
}
```

Which meeting going on increase count or decrease when it finished.

```
//using treeMap.=====
int minMeetingRooms_02(vector<vector<int>> &intervals)
{
    if (intervals.size() <= 1)
        return intervals.size();

    int n = intervals.size();
    map<int, int> bst;
    for (auto &ar : intervals)
    {
        bst[ar[0]]++;
        bst[ar[1]]--;
    }

    int ans = 0, rooms = 0;
    for (auto ele : bst)
    {
        rooms += ele.second;
        ans = max(ans, rooms);
    }
    return ans;
}
```

*from kth meeting start hogi
kth meeting end hogi uski counter.*