# 253. Meeting Rooms II Premium

Medium · Topics · Companies · Hint

Given an array of meeting time intervals `intervals` where `intervals[i] =` [$start_i$, $end_i$], return *the minimum number of conference rooms required.*

**Example 1:**

```
Input: intervals = [[0,30],[5,10],[15,20]]
Output: 2
```

**Example 2:**

```
Input: intervals = [[7,10],[2,4]]
Output: 1
```

**Constraints:**

- $1 <= \text{intervals.length} <= 10^4$
- $0 <= start_i < end_i <= 10^6$

Seen this question in a real interview before? 1/5

Yes No

Accepted 951.9K | Submissions 1.8M | Acceptance Rate 51.6%

## Topics

Array · Two Pointers · Greedy · Sorting · Heap (Priority Queue) · Prefix Sum

## Companies

0 - 3 months

| | |
|---|---|
| Google | 16 |
| Amazon | 12 |
| Microsoft | 8 |
| Bloomberg | 8 |
| TikTok | 7 |
| Meta | 4 |
| Uber | 4 |
| Netflix | 4 |
| Adobe | 2 |
| Splunk | 2 |

0 - 6 months

| | |
|---|---|
| Apple | 4 |
| Oracle | 4 |
| Snap | 2 |

6 months ago

| | |
|---|---|
| IBM | 15 |
| Turo | 6 |
| Yandex | 4 |
| eBay | 4 |
| Salesforce | 4 |
| Goldman Sachs | 3 |
| Pinterest | 3 |
| ServiceNow | 3 |
| Yahoo | 3 |
| Miro | 3 |

## Hint 1

Think about how we would approach this problem in a very simplistic way. We will allocate rooms to meetings that occur earlier in the day v/s the ones that occur later on, right?

## Hint 2

If you've figured out that we have to **sort** the meetings by their start time, the next thing to think about is how do we do the allocation?
There are two scenarios possible here for any meeting. Either there is no meeting room available and a new one has to be allocated, or a meeting room has freed up and this meeting can take place there.

## Hint 3

An important thing to note is that we don't really care **which** room gets freed up while allocating a room for the current meeting. As long as a room is free, our job is done.

We already know the rooms we have allocated till now and we also know when are they due to get free because of the end times of the meetings going on in those rooms. We can simply check the room which is due to get vacated the earliest amongst all the allocated rooms.

## Hint 4

Following up on the previous hint, we can make use of a min-heap to store the end times of the meetings in various rooms.

So, every time we want to check if any room is free or not, simply check the topmost element of the min heap as that would be the room that would get free the earliest out of all the other rooms currently occupied.

If the room we extracted from the top of the min heap isn't free, then no other room is. So, we can save time here and simply allocate a new room.

## Similar Questions

| | |
|---|---|
| **Merge Intervals** | Medium |
| **Meeting Rooms** 🔒 | Easy |
| **Minimum Number of Arrows to Burst Balloons** | Medium |
| **Car Pooling** | Medium |
| **Number of Flowers in Full Bloom** | Hard |
| **Meeting Rooms III** | Hard |
| **Total Cost to Hire K Workers** | Medium |
| **Points That Intersect With Cars** | Easy |

Discussion (42)