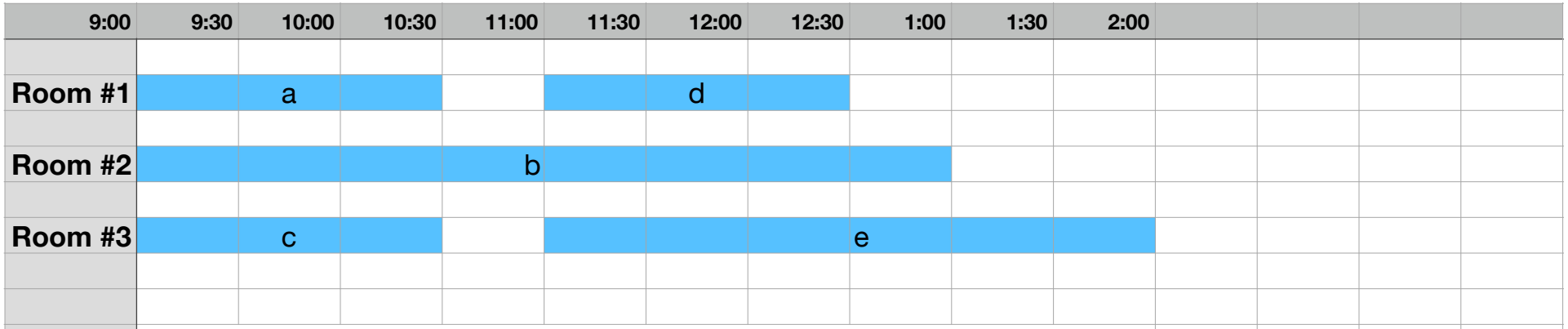
All Interval Scheduling Problem - Requests Sorted by Start Time



Suppose we need to add another classroom to accommodate the next interval.

What are the reasons that the algorithm add a new classroom?

It happens because the new interval (f here) starts BEFORE THE END of the last-scheduled interval in the classroom at the top of the priority queue (d here). The other last-scheduled intervals in the other classrooms end later than d (or at the same time), so the new interval starts before the end of ALL the last-scheduled intervals in ALL the classrooms. Not only that, the new interval starts after (or at the same time) as all the last-scheduled intervals in all the classrooms. This is true because the intervals are processed in order of increasing start time. The foregoing facts imply that the start time of the new interval is “in between” the start and finish times of all the last-scheduled intervals. Therefore there are points in the interior of the new interval that are common to all the last-scheduled intervals in all the existing classrooms. By the definition of the depth d then, d must be ≤ the existing number of classrooms plus one. Therefore when we add the new classroom, the new number of classrooms will not exceed d. This proves that the algorithm never uses more than d classrooms.