CS3230 : Tutorial - 1

Bakh Khoussainov

23/24-Aug-2011

1. Consider the interval scheduling problem.
    (a) Give an example of requests where the rule “select the available request that starts earliesst” does not yield an optimal solution.
    (b) Give an example of requests where the rule “select the available request with the smallest interval time” does not yield an optimal solution.

2. Consider the algorithm provided in the lecture that builds an optimal solution for the interval scheduling problem. The proof of its correctness uses induction. Write down explicitly (1) the base case of the induction, (2) The induction hypothesis, and (3) the inductive step.

3. Consider the interval partitioning problem. Recall that the depth is the maximum number of intervals that pass over one point in time. Write an algorithm (you might right a pseudo-code) that given n intervals computes the depth of the set of intervals.

4. Explain in a few (five or six at most) compact sentences why the algorithm provided in the lecture that solves the interval scheduling problem is a greedy algorithm.

5. Consider the algorithms (provided in the lecture) that solve the interval scheduling and the interval partitioning problem. In implementations of those algorithms one needs to sort requests by either their finishing time or starting time.
    (a) Which sorting algorithms would you use to sort the requests? (you need just to name some of the sorting algorithms that you know).
    (b) For the interval scheduling problem would you sort request by their finishing time or their starting time? Give a short ((4-5 sentences) explanation to your answer.
    (c) For the interval partitioning problem would you sort request by their finishing time or their starting time? Give a short ((4-5 sentences) explanation to your answer.

6. Consider the interval scheduling problem. Let $I_1, \ldots, I_n$ be the input set of intervals. For each of these intervals I the conflict number $n(I)$ is the total number of intervals J such that $I \cap J \neq \emptyset$. Suppose we select a schedule based on "take the interval with the smallest conflict number" rule and preserving the compatibility condition.

   (a) Would such a schedule give an optimal solution?

   (b) Explain your answer by proving either a proof or counter-example?