CS 2443: Examination 2

Department of Computer Science, IIT Hyderabad

15-Apr-2021

- Total marks: 15
- The term graph means simple undirected graphs without self loops and parallel edges.
- Maintain academic honesty.
- Kindly submit before the deadline and map answers properly in gradescope.

---

1. Consider a weighted version of the class scheduling problem, where different classes have different credits (totally unrelated to the duration of the class lectures). Your goal is to choose a set of non-conflicting classes that give you the largest possible number of credits, given arrays of start times, end times, and credits as input. Prove that the greedy algorithm described for scheduling described in the class (i.e., choose the class that ends first and recurse) does not always return an optimal schedule. [1]

2. Suppose we are given both an undirected graph G with weighted edges and a minimum spanning tree T of G. Describe an algorithm to update the minimum spanning tree when the weight of a single edge e is decreased. Your algorithm should modify T so that it is still a minimum spanning tree. Prove its correctness and analyze the running time. [3]

3. Let G be a digraph with arbitrary edge weights(which may be positive, negative, or zero), possibly with negative cycles, and let s be a given source vertex. We are also given a number $dist(v)$ for all v. Design an algorithm that should output one of the following correctly.

   (a) $dist(v)$ is indeed the shortest-path distance from s to v, for every vertex v.

   (b) Either condition (a) is false or there is a negative cycle in G.

   Prove the correctness of your algorithm and analyze the running time. [3]

4. Consider the execution Johnson's algorithm for All Pair Shortest Path on the graph H given in Figure 1. [4]

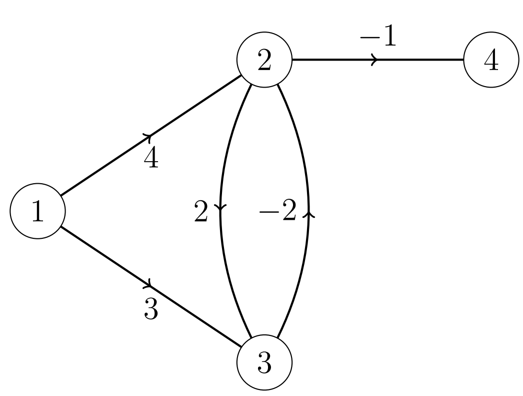

Figure 1: Graph H

5. Let X be a set of n intervals on the real line. A subset of intervals $Y \subseteq X$ is a *hitting set* of X if every interval in X intersects with an interval in Y. Describe and analyze an efficient algorithm (i.e., prove the correctness of the algorithm and running time) to compute a smallest cardinality hitting set of X. Assume that your input consists of two arrays $L[1 \ldots n]$ and $R[1 \ldots n]$, representing the left and right endpoints of the intervals in X. [4]