Name(last, first): _____

U C L A Computer Science Department

CS 180 Algorithms & Complexity UID : _____

Midterm **Total Time: 90 minutes**

Each problem has 20 points: 5 problems (upload ONE pdf that has at most 2 pages per problem to Gradescope).

For all 5 problems: algorithms, proofs and time complexity analysis should be described in bullet point format. Unless otherwise stated, you need to prove the correctness of your algorithm and you need to analyze its time complexity.

Problem 1: a. Describe Dijsktra's shortest path algorithm on a given weighted and undirected graph G=(V,E) **b.** Prove that it gives shortest paths **c.** Implement the algorithm using the heap structure.

Problem 2: Suppose that you are given an algorithm as a black box. You cannot see how it is designed. It has the following properties: If you input any sequence of real numbers, and an integer k, the algorithm will answer "yes" or "no," indicating whether there is a subset of the numbers whose sum is exactly k. Show how to use this black box to find the subset whose sum is k, if it exists. You should use the black box O(n) times (where n is the size of the sequence).

Problem 3. Let G =(V, E) be a connected weighted undirected graph, and let T be an MST of G. Suppose that we now add a new vertex v to G, together with some weighted edges from v to vertices of G. Design a linear-time algorithm to find a new MST that includes v.

Problem 4: Consider an undirected connected graph G=(V,E) where each vertex has an even degree. Design a linear-time algorithm that directs every edge (assigns a direction to each edge) such that the in-degree of every vertex is equal to the out-degree of that vertex. Analyze its time complexity (a proof is not needed).

Problem 5:
Show each step of DFS (Depth First Search) on this graph starting from vertex A.