Algorithms Final Exam

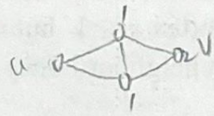
January 10, 2018, 09:10 - 12:10

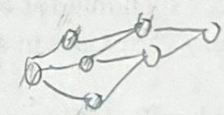
Please answer the following 7 questions on the answer sheets provided. Be sure to write your name and student ID on all answer sheets you use. You may bring one A4-sized, hand-written, double-sided "cheat sheet". No other books, notes, or calculators may be used during the exam.

If you want to use any result or theorem that has been taught in class (including homeworks), you may do so but you must state the result or theorem clearly before using it. You may assume that all arithmetic operations take $O(1)$ time.

Problem 1 (15%)

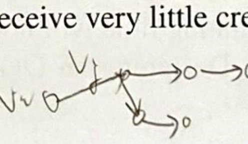
Given an undirected graph $G = (V, E)$ and two nodes $u, v \in V$, find the number of shortest paths from u to v. (In this problem, there is no edge cost, so shortest paths are the paths with the smallest number of edges.) Notice that you do not need to specify all shortest paths, just counting the number is sufficient. Your algorithm should run in time $O(V + E)$ in order to receive full credit. Briefly justify the correctness and analyze the running time.





Problem 2 (10%)

Given a directed graph $G = (V, E)$, the goal is to determine whether there exists two vertices v_i and v_j such that there is a path from v_i to v_j but no path from v_j to v_i. Design a $O(V + E)$-time algorithm to solve this problem. Briefly justify the correctness and analyze the running time. Algorithms with higher asymptotic running time will receive very little credits.



Problem 3 (15%)

There are n different locations $v_1, v_2, \dots, v_n$. An electric car wants to travel among these locations. Furthermore, charging stations only appear in some of these locations (and not on the roads between them). The car can only drive k kilometers after charging.

Given a directed graph $G = (V, E)$ with n nodes denoting the n different locations $v_1, v_2, \dots, v_n$, edges denoting all possible roads, and edge costs are the driving distances. All distances are positive integers. Given any location (node), checking whether there is a charging station at the location takes $O(1)$ time. Starting with a fully charged electric car at location v_1, design an $O(k(V + E))$-time algorithm to find all reachable locations. Briefly justify the correctness and analyze the running time.

Problem 4 (15%)

Given an undirected connected graph $G = (V, E)$ with each edge having a time-varying cost. The cost of edge e is $a_e t + b_e$ for every edge e at time t. Given a time interval $[0, M]$. The goal is to choose a spanning tree T and a time t such that $0 \le t \le M$ and constructing the whole tree T at time t minimizes the total cost among all possible choices of T and t. You must design an efficient algorithm in order to get the full credit, but any polynomial time algorithm gives partial credit. Briefly justify the correctness and analyze the running time.

For problems 5 and 6, you must either

1. design an algorithm with running time polynomial in $|V|$, briefly justify the correctness,

   or

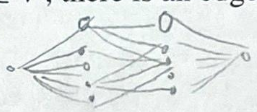
2. prove that the problem is NP-complete.

You may use the fact that SAT, 3-SAT, Vertex Cover, Set Cover, Independent Set, Hamiltonian Path and Hamiltonian Cycle problems are all NP-complete. Also, you do not need to prove that problems 5, 6 are in NP.

MST or Flow NPC

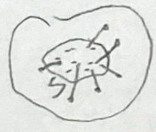
Problem 5 (15%)

Given an undirected bipartite graph $G = (V, E)$ and a constant k. Determine whether there exists a set S of k edges such that for every vertex $v \in V$, there is an edge $(u, v) \in S$.

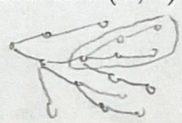


Problem 6 (15%)

Given an undirected graph $G = (V, E)$ and a constant k. Determine whether there exists a set S of k vertices such that for every vertex $v \in V$ but not in S, there is a vertex u in S such that $(u, v) \in E$. (i.e. every vertex not in S has a neighbor in S.)



V



Problem 7 (15%)

Given a graph $G = (V, E)$ with all edge costs between 1 and 4. Design an $O(V + E)$-time 2-approximation algorithm to find the shortest paths from a given vertex s to all other vertices v. Algorithms using longer running time will not receive any credit. Briefly justify the correctness and analyze the running time. Designing an $O(V + E)$-time 4-approximation algorithm gives partial credit.

Administrative issues:

1. The exam score and adjustment will be announced on 1/19(Fri) the latest.
2. You can check exam scores on 1/19(Fri) 9-11am at MD718.