CSCE629 Analysis of Algorithms

Homework 9

Texas A&M U, Fall 2019 *11/01/19*

Lecturer: Fang Song *Due: 10am, 11/08/19*

Instructions.

- Typeset your submission by LaTeX, and submit in PDF format. Your solutions will be graded on *correctness* and *clarity*. You should only submit work that you believe to be correct, and you will get significantly more partial credit if you clearly identify the gap(s) in your solution. You may opt for the "I'll take 15%" option.
- You may collaborate with others on this problem set. However, you must **write up your own solutions** and **list your collaborators and any external sources** for each problem. Be ready to explain your solutions orally to a course staff if asked.
- For problems that require you to provide an algorithm, you must give a precise description of the algorithm, together with a proof of correctness and an analysis of its running time. You may use algorithms from class as subroutines. You may also use any facts that we proved in class or from the book.
- **If you describe a Greedy algorithm, you will get no credit without a formal proof of correctness, even if your algorithm is correct.**

This assignment contains 4 questions, 4 pages for the total of 50 points and 0 bonus points. A random subset of the problems will be graded.

Exercises. Do not turn in.

1. (Max bipartite matching)
   (a) Give a linear-programming formulation of the bipartite maximum matching problem. The input is a bipartite graph $G = (U \cup V; E)$, where $E \subseteq U \times V$; the output is the largest matching in G. Your linear program should have one variable for each edge.
   (b) Now dualize the linear program from part (a). What do the dual variables represent? What does the objective function represent? What problem is this!?

Problems to turn in.

1. (15 points) (Vertex cover) A *vertex cover* of an undirected graph $G = (V, E)$ is a subset of the vertices which touches every edge—that is, a subset $S \subseteq V$ such that for each edge $u, v \in E$, one or both of u, v are in S. Describe and analyze an algorithm, as efficient as you can, to find a minimum vertex cover in a bipartite graph.

2. (Updating max flow) You are given a flow network $G = (V, E)$ with source s and sink t, and integer capacities.

   (a) (10 points) Suppose that you are given a max flow in G. Now we increase the capacity of a single edge $(u, v) \in E$ by 1. Given an $O(m + n)$-time algorithm to update the max flow.

   (b) (10 points) Now suppose all edges have unit capacity and you are given a parameter k. The goal is to delete k edges so as to reduce the maximum $s - t$ flow in G as much as possible. In other words, you should find a set of edges $F \subseteq E$ so that $|F| = k$ and the maximum $s - t$ flow in $G' = (V, E - F)$ is as small as possible subject to this. Describe and analyze a polynomial-time algorithm to solve this problem.

3. (15 points) (Filling classrooms) Faced with the threat of brutally severe budget cuts, Potemkin University has decided to hire actors to sit in classes as "students", to ensure that every class they offer is completely full. Because actors are expensive, the university wants to hire as few of them as possible.

   Building on their previous leadership experience at the now-defunct Sham-Poobanana University, the administrators at Potemkin have given you a directed acyclic graph $G = (V, E)$, whose vertices represent classes, and where each edge $i \to j$ indicates that the same "student" can attend class i and then later attend class j. In addition, you are also given an array $cap[1, \ldots, V]$ listing the maximum number of "students" who can take each class. Describe an analyze an algorithm to compute the minimum number of "students"that would allow every class to be filled to capacity.