# CS515: Algorithms and Data Structures, Fall 2019

## Homework 4*

## Due: Thr, 12/5/19

**Homework Policy:**

1. Students should work on homework assignments in groups of preferably three people. Each group submits to TEACH one set of typeset solutions, and hands in a printed hard copy in class or slides the hard copy under my door before the midnight of the due day. The hard copy will be graded.
2. The goal of the homework assignments is for you to learn solving algorithmic problems. So, I recommend spending sufficient time thinking about problems individually before discussing them with your friends.
3. You are allowed to discuss the problems with other groups, and you are allowed to use other resources, but you *must* cite them. Also, you must write everything in your own words, copying verbatim is plagiarism.
4. *I don't know policy:* you may write "I don't know" *and nothing else* to answer a question and receive 25 percent of the total points for that problem whereas a completely wrong answer will receive zero.
5. Algorithms should be explained in plain english. Of course, you can use pseudocodes if it helps your explanation, but the grader will not try to understand a complicated pseudocode.
6. More items might be added to this list. ☺

**Problem 1.** [35 pts] We can speed up the Edmonds-Karp 'fat pipe' heuristic, at least for integer capacities, by relaxing our requirements for the next augmenting path. Instead of finding the augmenting path with maximum bottleneck capacity, we find a path whose bottleneck capacity is at least half of maximum, using the following capacity scaling algorithm. The algorithm maintains a bottleneck threshold $\Delta$; initially, $\Delta$ is the maximum capacity among all edges in the graph. In each phase, the algorithm augments along paths from $s$ to $t$ in which every edge has residual capacity at least $\Delta$. When there is no such path, the phase ends, we set $\Delta \leftarrow \lfloor \Delta/2 \rfloor$, and the next phase begins.

(a) How many phases will the algorithm execute in the worst case, if the edge capacities are integers?

(b) Let $f$ be the flow at the end of a phase for a particular value of $\Delta$. Let $S$ be the nodes that are reachable from $s$ in the residual graph $G_f$ using only edges with residual capacity at least $\Delta$, and let $T = V \backslash S$. Prove that the capacity (with respect to $G$'s original edge capacities) of the cut $(S, T)$ is at most $|f| + E \cdot \Delta$.

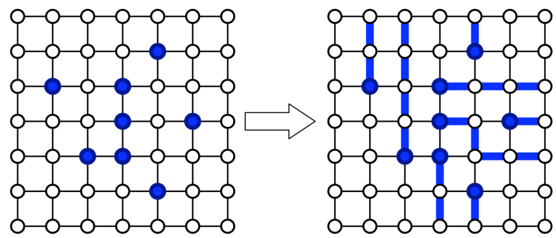
*Problems are from Jeff Erickson's lecture notes. Looking into similar problems from his lecture notes on maximum flow, their applications, and linear programming is recommended.

(c) Prove that in each phase of the scaling algorithm, there are at most $2E$ augmentations.

(d) What is the overall running time of the scaling algorithm, assuming all the edge capacities are integers?

**Problem 2.** [30 pts] An $n \times n$ grid is an undirected graph with $n^2$ vertices organized into $n$ rows and $n$ columns. We denote the vertex in the $i$th row and the $j$th column by $(i, j)$. Every vertex in the grid have exactly four neighbors, except for the boundary vertices, which are the vertices $(i, j)$ such that $i = 1$, $i = n$, $j = 1$, or $j = n$.

Let $x_1, x_2, \ldots, x_k$ be distinct vertices, called terminals, in the $n \times n$ grid. The quick escape problem is to find $k$ vertex-disjoint paths in the grid that connect the terminals to any $k$ distinct boundary vertices. Describe and analyze an efficient algorithm to solve the escape problem. Prove that your algorithm is correct.



**Problem 3.** [35 pts] Given points $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ in the plane, the linear regression problem asks for real numbers $a$ and $b$ such that the line $y = ax + b$ fits the points as closely as possible, according to some criterion. The most common fit criterion is minimizing the $\ell_2$ error, defined as follows:

$$\varepsilon_2 = \sum_{i=1}^{n} (y_i - (ax_i + b))^2.$$

But there are several other fit criteria, some of which can be optimized via linear programming.

(a) The $\ell_1$ error (or total absolute deviation) of the line $y = ax + b$ is defined as follows:

$$\varepsilon_1 = \sum_{i=1}^{n} |y_i - (ax_i + b)|.$$

Describe a linear program whose solution (a, b) describes the line with minimum $\ell_1$ error.

(b) The $\ell_\infty$ error (or maximum absolute deviation) of the line $y = ax + b$ is defined as follows:

$$\varepsilon_\infty = \max_{i=1}^{n} |y_i - (ax_i + b)|.$$

Describe a linear program whose solution (a, b) describes the line with minimum $\ell_\infty$ error.

**Problem 4.**

(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!?