COMP 360 - Winter 2016 - Assignment 2

Due: 6pm Feb 10th.

General rules: In solving these questions you may consult books but you may not consult with each other. You should drop your solutions in the assignment drop-off box located in the Trottier Building.

"Is this going to be on the midterm?"

---

1. (10 Points) Consider the following two algorithms for finding the maximum flow:

Algorithm 1: Scaling max-flow

- Initially set $f(e) := 0$ for all edges e.
- Set Δ to be $\max c_e$ rounded down to a power of 2.
- While $\Delta \geq 1$:
- While there is an s,t-path P in $G_f(\Delta)$:
- Augment the flow using P and update $G_f(\Delta)$.
- Endwhile.
- Set $\Delta := \Delta/2$.
- Endwhile.
- Output f.

Algorithm 2: The fattest path algorithm

- Initially set $f(e) := 0$ for all edges e.
- While there exists an s,t-path in G_f:
- Augment the flow using the fattest s,t-path P in G_f.
- Update G_f.
- Endwhile.
- Output f.

In the second algorithm the fattest means the largest bottleneck. From the class we know that the number of augmentations in Algorithm 1 is at most $2m\lceil \log_2 K \rceil$, where K is the maximum capacity of an edge. Deduce from this that the number of augmentations in Algorithm 2 is also at most $2m\lceil \log_2 K \rceil$.

2. (35 points) A *key edge* of a flow network is an edge whose deletion causes the largest drop in the value of the maximum flow. Let f be an arbitrary maximum flow. True of False (Prove or Disprove)?

   (a) A key edge is always an edge with maximum capacity.

   (b) Deletion of a key edge e decreases the value of the maximum flow by the capacity of e.

   (c) Deletion of a key edge e decreases the value of the maximum flow by $f(e)$.

   (d) A key edge e in a minimum cut (A, B) has the maximum value of f among edges belonging to that cut (edges from A to B).

   (e) An edge that does not belong to some minimum cut is not a key edge.

3. (15 points) Use the Max-Flow problem to solve the following problem. The input is a number k and a *bipartite* graph $G = (V, E)$ where some of the edges are colored red. We would like to find out whether we can color more edges red (if necessary) so that every vertex is incident to exactly k red edges.

4. (20 points) Show that for every flow network, there exists an execution of the Ford-Fulkerson algorithm that never decreases the value of the flow on any of the edges (i.e. never "pushes back" the flow on any of the edges).

5. (20 points) Suppose that the set $\{1, \ldots, m\}$ is partitioned into disjoint sets $S_1, \ldots, S_k$. Also let $a_1, \ldots, a_t$ be positive integers with $a_1 + \ldots + a_t = m$. Given $S_1, \ldots, S_k$ and $a_1, \ldots, a_t$, construct a flow network with the following property: The network has maximum flow m if and only if there is a partition $T_1, \ldots, T_t$ of $\{1, \ldots, m\}$ such that

   (a) $|T_i| = a_i$ for all $i = 1, \ldots, t$;

   (b) The elements of every S_i belong to different T_j's (That is if $a, b \in S_i$ for some i, then $a \in T_x$ and $b \in T_y$ for some $x \neq y$).

   (Clarification: This is one question. Both conditions (a) and (b) must hold at the same time.)