CS 573: Graduate Algorithms, Fall 2011
HW 4 (due in class on Tuesday, November 1st)

This homework contains five problems. **Read the instructions for submitting homework on the course web page**. In particular, *make sure* that you write the solutions for the problems on separate sheets of paper. Write your name and netid on each sheet.

Collaboration Policy: For this home work students can work in groups of up to three students each. Only one copy of the homework is to be submitted for each group. Make sure to list all the names/netids clearly on each page.

Note on Proofs: Details are important in proofs but so is conciseness. Striking a good balance between them is a skill that is very useful to develop, especially at the graduate level.

1. (20 pts) Sampling is a powerful methodology in randomized algorithm design although we did not get to see it formally yet. The idea is that a small random sample of the input retains many interesting properties of the original input with good probability; one can then run an algorithm on the sample instead of the original input. This problem gives an example. Let S be a set of n numbers (assume they are distinct) and say you want to approximate their median. A number x is an ϵ-approximate median of S if at least $(\frac{1}{2} - \epsilon)n$ numbers in S are less than x and at least $(\frac{1}{2} - \epsilon)n$ are greater than x. Consider the following algorithm. Pick a subset $S' \subseteq S$ uniformly at random and return the median of the sample S' as the approximate median. Show that there is a constant c *independent* of n such that if the sample size $|S'| \geq c$ then the output is a 0.05-approximate median with probability at least 0.99. The sampling can be done with or without replacement from S; by the term "with replacement" we mean that one can picks numbers independently c times with the possibility of picking the same number multiple times (in which case the median is computed for the resulting multiset). Give an expression for c as a function of ϵ and δ that ensures that the sampling algorithms outputs an ϵ-approximate median with probability at least $(1 - \delta)$.

2. (20 pts) Consider the balls and bins experiment in which m balls are thrown into n bins where each ball is thrown in to a bin chosen uniformly at random. We showed in lecture, via Chernoff bounds, that when $m = n$ the expected value of the maximum bin load is $O(\log n / \log \log n)$ with high probability; this can be generalized to show that the expected value of the maximum load is $O(\frac{m}{n} \log n / \log \log n)$ with high probability. Note that the expected load of any bin is $\frac{m}{n}$ and hence the bound is showing that the maximum load is at most a $\log n / \log \log n$ *multiplicative* factor away. One can show a related bound that is useful when m is large. Show that, for any fixed $\epsilon > 0$, the value of the maximum is at most $(1+\epsilon)\frac{m}{n} + c \cdot \frac{\log n}{\epsilon}$ with probability at least $1 - 1/n$ for some suitably large universal constant c. Do you see the advantage of this bound when $m > n \log n$? *Hint:* Use Chernoff bound.

3. (20 pts) Let $G = (V, E)$ be a flow network with integer capacities $c : E \to \mathbb{Z}_+$ and $s, t \in V$ be the source and sink nodes. Suppose you computed a maximum integer flow $f : E \to \mathbb{Z}_+$. Now you want to recompute the maximum flow for a modified network in which the capacity of a given edge $e \in E$ has been *reduced* by k units. Show how this can be done in $O(k(m + n))$-

time. You may want to first think about the easier case when the capacity of e is *increased* by k units.

4. (20 pts) Let $G = (V, E)$ be a flow network with integer capacities $c : E \to \mathbb{Z}_+$ and $s, t \in V$ be the source and sink nodes. Recall that a partition (A, B) of V is an s-t cut if $s \in A$ and $t \in B$.

   - Give an example of a graph in which the number of distinct s-t minimum cuts is exponential in n the number of nodes.
   - An s-t cut (A, B) is said to be a *minimal* minimum cut if there is no other cut (A', B') such that $A' \subset A$. It is not obvious that there exists a *unique* minimal min-cut. However, show that it exists by proving following: if (A, B) and (A', B') are both minimum s-t cuts then so is $(A \cap A', V - (A \cap A'))$. In fact show that the set of reachable nodes from s in the residual graph G_f of a maximum flow f is a unique minimal min-cut.
   - Extend the above show that there is a unique *maximal* mincut?
   - Use the above to describe an efficient algorithm to find whether a given graph G has a *unique* minimum s-t cut.

5. (20 pts) The Computer Science Department at UIUC has n professors. They handle department duties by taking part in various committees. There are m committees and the j'th committee requires k_j professors. The head of the department asked each professor to volunteer for a set of committees. Let $S_i \subseteq \{1, 2, \ldots, m\}$ be the set of committees that professor i has volunteered for. A committee assignment consists of sets $S'_1, S'_2, \ldots, S'_n$ where $S'_i \subseteq \{1, 2, \ldots, m\}$ is the set of committees that professor i will participate in. A committee assignment has to satisfy two constraints: (i) for each professor i, $S'_i \subseteq S_i$, that is each professor is only given committees that he/she has volunteered for, and (ii) each committee j has k_j professors assigned to it, or in other words j occurs in at least k_j of the sets $S'_1, S'_2, \ldots, S'_n$. The head of the department notices that often there is no valid committee assignment because professors are inclined to volunteer for as few committees as possible. To overcome this, the definition of a valid assignment is relaxed as follows. Let ℓ be an integer. An assignment $S'_1, S'_2, \ldots, S'_n$ is said to be valid if (i) $|S'_i - S_i| \leq \ell$ and (ii) each committee j has k_j professors assigned to it. The modified condition (i) means that a professor i may be assigned up to ℓ committees not on the list S_i that he/she volunteered for. Describe an algorithm to check if there is a valid committee assignment with the relaxed definition.

Questions to ponder: Many!

- Do you understand why using 2-universal hashing gives us $O(1)$ expected time per operation if the number of items stored in the table is at most the size of the table and we only do insertions and lookups?
- Read the proof of the Schwartz-Zippel lemma.
- We saw the application of Schwartz-Zippel lemma for checking whether a bipartite graph has a perfect matching. See the wikipedia page to see how to apply it to matching in general graphs.

- We saw network flow with capacities on the edges. Suppose we have capacities on the nodes as well as edges. Can you reduce it to the case with only edge capacities. See Kleinberg-Tardos Problem 7.13.
- See Kleinberg-Tardos book, Chapter 7, for many interesting problems on network flows and its applications. Problems 11, 12, 13, 14, 17, 18, 32, 33 are particularly useful to look at.