NATIONAL UNIVERSITY OF SINGAPORE
SCHOOL OF COMPUTING

Midterm Assessment for CS5330 – Randomized Algorithms

3 March 2020 Time Allowed: 180 minutes

The outline of solutions for the midterm exam questions are in sans serif font below.

QUESTIONS:

1. **(20 points)** A random variable X takes values over the non-negative integers. X satisfies $\mathbb{E}[X] = 1$ and $\mathsf{Var}[X] = 2$. Decide if each of the following **must** be true about X. If the statement is true, write why. If it is false, show a counterexample.

   (a) $\Pr[X \geq 5] \leq \frac{1}{5}$
   True: Follows from Markov.

   (b) $\Pr[X \geq 2] > 0$
   True: Suppose $\Pr[X \geq 2] = 0$. Then, X can only take values over 0 and 1, and so, $\mathsf{Var}[X] \leq \mathbb{E}[X^2] \leq 1$.

   (c) $\Pr[X = 1] > 0$
   False: Suppose X is 0 with probability $\frac{2}{3}$ and 3 with probability $\frac{1}{3}$.

   (d) $\Pr[X \leq 1] \leq \Pr[X \geq 1]$.
   False: Suppose X is 0 with probability $\frac{2}{3}$ and 3 with probability $\frac{1}{3}$.

2. **(10 points)** In the fictional game *mossix*, you repeatedly toss a (fair, six-sided) die. If it ever turns up to be an odd number, you lose. Otherwise, your score is the number of tosses that it takes you to toss the first '6'. What is the probability that your score is 1 conditioned on the event that you don't lose? Explain your reasoning.

   This question is inspired by a puzzle due to Elchanan Mossel described here. The probability of tossing a 6 on the first toss is clearly $1/6$. The probability of not losing is $\sum_{i\geq 0} \left(\frac{2}{6}\right)^i \cdot \frac{1}{6}$ where the i'th term in the sum is the probability that you toss 2 or 4 in the first $i-1$ tosses followed by a 6; the sum is equal to $\frac{3}{2} \cdot \frac{1}{6} = \frac{1}{4}$. Hence, the conditional probability is $(1/6)/(1/4) = 2/3$. The answer may seem a bit of a paradox because in a 3-sided die with 2, 4 and 6, the probability of getting a 6 in the first toss is $1/3$. Intuitively, what's going on is that the condition of not losing biases the number of throws to be small.

3. **(20 points)** Recall Karger's algorithm that repeatedly contracts random edges.

   (a) **(5 points)** Why does Karger's algorithm return the min-cut with probability 1 when it's run on trees?
   The graph remains a tree after contraction along any edge.

   (b) **(15 points)** Suppose we modify Karger's algorithm to contract random pairs of vertices at each step (instead of random edges). Describe a graph on n vertices for which this modified algorithm returns a minimum cut with probability $2^{-\Omega(n)}$. Explain why.
   Consider two cliques A and B of size $n/2$ each with an edge between them. The mincut size is 1. Note that the algorithm's output is 1 if and only if at each step, two vertices from the same clique are contracted. Suppose at some iteration, there are i vertices in the left clique and j in the right. Then, probability of a random pair being from the same clique is $\left(\binom{i}{2}+\binom{j}{2}\right)/\binom{i+j}{2} = 1 - ij/\binom{i+j}{2} \leq 1 - 2ij/n^2$. If $i, j \geq n/3$, then this fraction is $\leq 7/9$. So, for the first $n/6$ contractions, the probability of choosing both vertices from the same clique is $\leq 7/9$. Hence, the probability of the algorithm outputting the mincut is $\leq (7/9)^{n/6} = 2^{-\Omega(n)}$.

4. **(15 points)** Suppose $G = (V, E)$ is an undirected graph with vertex set V and edge set E. Let $n = |V|$ and $m = |E|$.

   Order the vertices in V randomly. Let I consist of the set of vertices v whose neighbors all occur after v in the order. Why is I an independent set (i.e., no edges between vertices in I)? Show that:

$$\mathbb{E}[|I|] = \sum_{v \in V} \frac{1}{\deg(v)+1}$$

   where $\deg(v)$ is the degree of vertex v.
   Suppose u and v are endpoints of an edge, and suppose $u \in I$. Then, v must occur later than u in the order, and therefore, $v \notin I$. Hence, I is an independent set.

   To calculate $\mathbb{E}[|I|]$, let X_v indicate if $v \in I$. Then, $\mathbb{E}[X_v]$ is the probability that v occurs earlier than any of its neighbors in the order. Since v and its neighbors are ordered randomly, the probability that v comes first is exactly $1/(\deg(v)+1)$. The result follows from linearity of expectations.

5. **(15 points)** Your friend runs up to you and breathlessly exclaims that she has found a randomized algorithm that compresses by 10% (in expectation) any binary string of length 10^6. She claims that her compression scheme is such that the original string is always recoverable from the compression. Prove to her that her claims cannot be true by using Yao's minimax principle.
   Consider the input distribution U that is uniform over all $n = 10^6$ bit strings. Let A be any fixed deterministic compression algorithm. There must be 2^n strings that are output by A because each n-bit string has its distinct compression. Since $\sum_{i=0}^{0.95n} 2^i \leq 2^{0.96n}$ strings have length $\leq 0.95n$, $\mathbb{E}_{x \sim U}[|A(x)|] \geq (1 - 2^{-0.04n})0.95n \geq 0.91n$. Thus, by the Yao minimax principle, for any randomized compression algorithm, the expected worst case length of the compression of an n-bit string is $\geq 0.91n$.

6. **(20 points)** You are in front of a huge playpen filled with balls of different colors. These balls are of n colors, and you want to estimate p_i, the fraction of balls in the pen with color i.

You sample m balls, each uniformly at random and independently. Let $q_i^{(m)}$ be the fraction of balls in the sample colored i.

(a) **(6 points)** Show that if $m = O(\varepsilon^{-2} \log \delta^{-1})$, with probability at least $1 - \delta$, $|q_1^{(m)} - p_1| \leq \varepsilon$.
Use the Hoeffding bound. Let X_i be the event that the j'th sampled ball has color 1. Then $\Pr[|\sum_j X_j - p_1 m| > \varepsilon m] \leq e^{-\Omega(\varepsilon^2 m)}$. Taking $m = C\varepsilon^{-2} \log \delta^{-1}$ for a large enough constant C makes this probability $\leq \delta$.

(b) **(7 points)** Show that if $m = O(\varepsilon^{-2}(n + \log \delta^{-1}))$, with probability at least $1 - \delta$:

$$\max_{S \subseteq [n]} \left| \sum_{i \in S} q_i^{(m)} - \sum_{i \in S} p_i \right| \leq \varepsilon.$$

Again use the Hoeffding bound. Fix any set S. Let X_j indicate the event that j'th sampled ball has a color from S. Taking $m = C\varepsilon^{-2} \log(2^n/\delta)$ for a large enough constant C ensures that $\Pr[|\sum_{j \in [m]} X_j - m \sum_{i \in S} p_i| > \varepsilon m] \leq \delta/2^n$. Finally, you can take a union bound over all 2^n possible S's.

(c) **(7 points)** Show that if $m = O(\varepsilon^{-2} \log(n\delta^{-1}))$, with probability at least $1 - \delta$:

$$\max_{i \in [n]} |q_i^{(m)} - p_i| \leq \varepsilon.$$

This is similar to part (a). By taking $m = C\varepsilon^{-2} \log(n/\delta)$ for a large enough constant c and using the Hoeffding bound, we get that for any fixed i, with probability at least $1 - \delta/n$, $|q_i^{(m)} - p_i| \leq \varepsilon$. Doing a union bound over the n possible values of i completes the proof.

(d) **[Optional Bonus! (+10 points)]** For part (c), actually $m = O(\varepsilon^{-2} \log \delta^{-1})$ samples suffice, without any dependence on n. Can you show this?
Use the Chernoff bound. For any fixed i, we get that $\Pr[|q_i - p_i| > \varepsilon] \leq e^{-\Omega(\varepsilon^2 m/p_i)}$. By the union bound, $\Pr[\max_i |q_i - p_i| > \varepsilon] \leq \sum_i e^{-\Omega(\varepsilon^2 m/p_i)}$. You can check that the function $F(p_1, \ldots, p_n) = \sum_{i=1}^n e^{-C\varepsilon^2 m/p_i}$ with the constraint $\sum_{i=1}^n p_i = 1$ is maximized when p_i is 1 for one i and 0 for all else. Therefore, $F(p_1, \ldots, p_n) \leq e^{-C\varepsilon^2 m}$ and so the result follows.