CS:5360 Fall 2018 Homework 4

Due: Thu, 10/25

Notes: (a) It is possible that solutions to some of these problems are available to you via textbooks on randomized algorithms or on-line lecture notes, etc. If you use any such sources, please acknowledge these in your homework fully *and* present your solutions in your own words. You will benefit most from the homework, if you seriously attempt each problem on your own first, before seeking other sources. (b) As mentioned in the syllabus, it is okay to form groups of two in solving and submitting homework solutions. But, my advice from (b) still applies: you will benefit most from the homework, if you seriously attempt each problem on your own first, before seeking help from your group partner. (c) Discussing these problems with any of your classmates is okay, provided you and your classmates are not being too specific about solutions. In any case, make sure that you take no written material away from these discussions *and* (as in (b)) you present your solutions in your own words. When discussing homework with classmates please be aware of guidelines on "Academic Dishonesty" as mentioned in the course syllabus.

1. Let $X_1, X_2, \ldots, X_n$ be a sequence of mutually independent binary random variables and let $X = \sum_{i=1}^{n} X_i$. Suppose that $E[X] = \ln^2 n$. Find as small a value of t as you can using a Chernoff bound to ensure that $Pr(X \geq E[X] + t) \leq \frac{1}{n}$.

2. A *graph coloring* is an assignment of colors to vertices of a graph so that no two adjacent vertices have the same color.

   Here is a simple algorithm to color the vertices of a graph $G = (V, E)$ using 3Δ colors, where Δ is the maximum degree of a vertex in G. Initially, let $C_v = \{1, 2, \ldots, 3 \cdot \Delta\}$ be the set of colors available for each vertex v. Every vertex is initially colorless. In each round, each colorless vertex v in G picks a color from C_v uniformly at random. Each vertex v that has just picked a color then checks if any neighbor has chosen the same color. If there is a neighbor who has chosen the same color as v, then v reverts back to being colorless. Otherwise, v makes its color choice permanent. Finally, for each colorless v, we delete from C_v all colors that became permanent at a neighbor of v. The algorithm then proceeds to the next round.

   This algorithm work for graphs in general, but here you need to analyze it only for trees. Also, you only need to analyze the first round of the algorithm. Specifically, show that if the algorithm is run on an n-vertex tree then with probability at least $1 - 1/n$, after one round of coloring, the graph induced by colorless vertices has maximum degree $3\Delta/4$. You may assume that $\Delta \geq 100 \ln n$.
   Hint: Calculate the probability that a vertex gets colored in the first round. Then, for each vertex v, calculate the expected number of neighbors of v that remain colorless at the end of the round. Finally, use Chernoff bounds to prove the claim.

3. Let $X_1, X_2, \ldots, X_n$ be independent, geometrically distributed random variables each with mean 2. Let $X = \sum_{i=1}^{n} X_i$ and $\delta > 0$. Obtain a Chernoff bound on $Pr(X \geq (1 + \delta) \cdot 2n)$ using the type of proof that we used to derive the Chernoff bound on the sum of binary random variables. Use your Chernoff bound to show that $Pr(X \geq 3n) \leq 1/c^n$ for constant $c > 1$.

4. The following randomized routing protocol for the problem of permutation routing on hypercubes was suggested in class.

> Suppose that instead of fixing bits in order from 1 to n, each packet chooses a random order (independent of other packets' choices) and fixes the bits in that order.

This seem like a natural way to randomize the deterministic bit-fixing protocol that performs poorly. Surprisingly, this randmized protocol is not much better, at least asymptotically. Show that there is an input for which this algorithm requires $2^{\Omega(n)}$ steps with high probability.
