Final Exam

Electronic submission via Gradescope, due 11:59pm Thursday 12/13.

You must work alone. You may consult the textbook/course notes, and other online resources (wikipedia, etc.). You may not discuss any aspect of this test with other students (whether or not they are enrolled in the class). You may NOT troll the internet looking for problem-specific solutions, or post course-related questions on online forums, with the exception of our class Piazza forum which you may use to ask clarification-type questions.

Please answer the questions thoroughly yet succinctly. Points may be deducted for needlessly convoluted solutions. As always, if there is a hole in your proof, fewer points will be deducted if you acknowledge that you are aware of this hole, versus trying to hide it.

1. For each of the following questions, provide a one to three sentence explanation.

(a) (3 points) Suppose you wish to bound the probability that a real-valued random variable exceeds its expectation by a significant amount. Describe, in general terms, when you would apply Markov’s inequality, versus when you would expect a better bound by applying Chebyshev’s inequality. Feel free to clarify your response with an illustrative example.

(b) (3 points) Describe the high-level steps in showing that some object (e.g. a graph with a certain property) exists, via the probabilistic method.

(c) (3 points) Suppose I give you a randomized algorithm $A$ for solving Problem X. [Recall that a randomized algorithm is one that, during its execution, is allowed to request uniformly random bits.] Suppose I tell you that the algorithm will only ask for at most 10 random bits during any execution (ie. the algorithm will only ask for 10 coin flips), and that for any input, I can (upper) bound the probability that the algorithm will fail by $10^{-10}$. Describe how to turn algorithm $A$ into a deterministic algorithm, and explain why your deterministic algorithm will always be correct.\footnote{The setup of this problem might sound bizarre, but this turns out to arise in some settings where you are trying to derandomize a randomized algorithm. For example, suppose you have an algorithm that uses 100 random bits, and fails with small probability. Now, suppose you can show that, in fact, your analysis of the failure probability does not require that the bits actually be independent. For example, suppose you can show that the failure probability analysis still goes through even if you just pick 10 random bits, and feed that same sequence in 10 times to make a string of 100 bits. Well, now the randomized algorithm only actually requires 10 random bits, and the bound on its failure probability remains unchanged. Now you could apply your solution to the problem to fully de-randomize it.}

(d) (4 points) Suppose we toss $n$ balls independently into $n$ bins. Describe two different approaches for proving that the number of bins with more than three balls will be tightly concentrated about its expectation (here “tightly concentrated” means that the probability that it is more than $c$ standard deviations from its expectation will be inverse exponential in $c$). For each of the approaches, just describe the high-level strategy/outline, no need to give any calculations.
(e) (3 points) Suppose your model of how people recognize faces is the following: upon seeing a face, you map the face to a $d$-dimensional vector of measurements (perhaps corresponding to the distance between the eyes, length sum of the length of the nose and width of the lips, the ratio of eye-brow angle to forehead protrusion, etc.), and then essentially look at how similar that vector is to each of the vectors corresponding to all the people whose faces you recognize. How large of a value of $d$ would be reasonable if people are able to recognize $n = 10,000$ faces? What value of $d$ would seem like overkill (assuming that humans do face-recognition in a not-awful manner). Justify your answer with a brief explanation, ideally referencing some specific result from this class.

(f) (3 points) Prove that any irreducible finite Markov chain that has at least one self-loop (that occurs with nonzero probability) must be aperiodic.

(g) (3 points) Describe an irreducible aperiodic Markov chain whose states correspond to the numbers 1 through 10, and whose stationary distribution assigns probability proportional to $i$ for state $i$ (i.e. the stationary distribution puts twice as much probability on state 2 than on state 1, etc.). [There are many correct answers, though try to describe a simple one : )] No proof necessary, just describe a Markov chain with the desired properties.

(h) You go to Vegas, and your friend is trying to convince you that there is a fancy scheme for deciding how much to bet on each round of roulette, and a rule for deciding when to stop playing, with the property that with probability 1 you will eventually stop playing (namely, the probability that you will have finished after $t$ rounds tends to 1 as $t$ tends to infinity), and your expected winnings when you stop playing is positive. [For this problem, assume that for each round of roulette, no matter what you bet on or how much you bet, your expected earnings are at most zero.]

i. (1 point) Describe such a strategy.

ii. (3 points) Explain why it is actually impossible for you to faithfully execute any/every strategy that has the properties described in the problem statement.

2. Suppose there are a bunch of people, and each person would like an email address. Suppose each person submits a list of 15 desired email addresses, and these have the property that no email address is in more than 3 peoples’ list.

(a) (6 points) Prove that it is possible to assign each person an email address on their list (without assigning any two people to the same address). [Hint: If you use the LLL, be sure to clearly define a distribution, define the bad events, and define a valid dependency graph. To this end, there are two natural choices of distribution—either assigning each email address to one of the $\leq 3$ people who list it, or having each person choose an email uniformly at random from their lists. One of these two distributions will end up working out more easily than the other.]

(b) (2 points) Describe an algorithm for finding a valid assignment of emails, which runs in time polynomial in the number of people. [For full credit, just describe the algorithm, no need to prove its correctness or a bound on the runtime.]

3. Consider the Markov chain corresponding to a lazy random walk on a cycle graph with $n$ nodes: specifically, the states correspond to the numbers $0, 1, \ldots, n - 1$, and at each timestep,
given that the chain is at state \( i \), with probability \( \frac{1}{2} \) the chain stays put, with probability \( \frac{1}{4} \) the chain then transitions from state \( i \) to state \( i+1 \mod n \), and with the remaining probability \( \frac{1}{4} \) the chain transitions from state \( i \) to \( i-1 \mod n \).

(a) (2 points) Define a coupling of two copies of this Markov chain, \( \{X_t\} \) and \( \{Y_t\} \), with the property that the distance between the two chain never changes by more than \( 1 \) at each time. Justify why your coupling is a valid coupling in at most one sentence.

(b) (5 points) Prove that the mixing time of the chain is \( O(n^2) \). [Hint: it might be helpful to analyze the distance between the two chains, namely \( X_t - Y_t \).]

(c) (1 points) Prove that the mixing time of this chain is at least \( cn^2 \) for some constant \( c \).

4. Suppose Stanford’s freshman class has \( n \) students, and the first name of each student was drawn independently from some unknown distribution \( D \) (corresponding to the distribution of first names that were popular in the year 1999, for example). In this question we will consider the total number of distinct first names in the freshman class. [E.g. if the class has four people, with names (Annie, Neha, Greg, Annie), then there are 3 distinct names.]

(a) (2 points) Suppose we put the students in a random order, and let \( A_i \) denote the event that the \( i \)th student has a name that is distinct from the first \( i-1 \) students. Are the events \( \{A_i\} \) independent? Explain with at most one sentence.

(b) (6 points) Prove that the number of distinct first names in the freshman class will be tightly concentrated about its expectation. Specifically, letting \( X \) denote the number of distinct first names in the class, and \( E[X] \) denote the expected number of distinct names given \( n \) independent draws from \( D \), prove that \( Pr[|X - E[X]| > \lambda] \leq 2e^{-\frac{\lambda^2}{2n}} \). [Hint: Azuma-Hoeffding!! Note that you don’t need to know what \( E[X] \) is prove that \( X \) is tightly concentrated about its expectation.]

5. Suppose there is a population of \( n \) yeast cells. At time \( t = 0 \), \( m \) of them have mutation \( A \). Suppose that at each timestep, the number of mutated cells will either increase or decrease by \( 1 \), with probability \( 1/2 \) each, until “fixation”—namely when either there are no mutated cells, or they are all mutated. Let \( T \) denote the first time when either all cells are mutated, or all cells are not mutated. Let \( \{X_t\} \) denote the number of mutated cells at time \( t \). In class we showed that \( E[T] = m(n - m) \), using the martingales \( \{X_t\} \) and \( \{X_t^2 - t\} \) and the stopping theorem. In this problem, we will compute the expected time until fixation, conditioned on ending up with all cells having the mutation. [Many versions of this problem arise in practical settings: e.g. we know the random process, the initial configuration, and the final outcome, and we want to estimate how much time must have passed given this information.]

(a) (3 points) Prove that \( Y_t = X_t^3 - 3t \cdot X_t \) is a martingale.

(b) (1 points) Briefly explain why we CANNOT apply the stopping theorem that we saw in class to \( \{Y_t\} \).

(c) (4 points) Even though \( Y_t \) does not satisfy the conditions for the version of the stopping theorem that we saw in class, a slightly more general stopping theorem does apply, and guarantees that \( E[Y_T] = E[Y_0] \). Using this fact, compute \( E[T|X_T = n] \) (namely, the expected time until all cells are the same, conditioned on ending up with all cells having the mutation).
6. [BONUS 3 points] [This is the same bonus as was on the midterm—no one got any points for it on the midterm so it seemed fair to ask it again.] Let $A$ denote a randomized algorithm, which always outputs the correct answer and has expected runtime 1 time-unit. In this problem, assume that you can call algorithm $A$, and run it for any specified amount of time (i.e. you can run it for exactly 1.3 time units, if you like). Describe an algorithm which calls $A$ as a subroutine, has runtime at most 100, and succeeds with probability strictly more than $1 - e^{-100/e}$. Prove that your algorithm has the desired properties. [Hint: If you simply run $A$ for at most $r$ steps, and then repeat until time 100, your probability of success is at least $1 - (1/r)^{100/r}$ (ignoring the rounding issue if $100/r$ is not an integer) which is maximized when $r = e \approx 2.72$. To do better, think about the behavior of algorithm $A$ that would make this analysis tight, and then modify the approach to exploit this.] [Grading note: 1 bonus point for any solution that is strictly better than $1 - e^{-100/e}$. For 2 bonus points, solution must be near-optimal. For 3 bonus points, must prove why proposed solution is near optimal.]