CS–174 Combinatorics & Discrete Probability, Fall 98

Sample Final Examination

12:30–3:30pm, 15 December

Read these instructions carefully

1. This is a **closed book** *exam. Calculators* **are** *permitted.*

2. This exam consists of 15 questions. The first ten questions are multiple choice; the remaining five require written answers.

3. Answer the multiple choice questions by circling the correct answer (or the best answer if more than one is correct). You should be able to answer all of these from memory, by inspection, or with a very small calculation. Incorrect answers attract a negative score, so if you do not know the answer **do not** *guess.*

4. Write your answers to the other questions in the spaces provided. None of these questions requires a long answer, so you should have enough space; if not, continue on the back of the page and state clearly that you have done so. **Show all your working**.

5. The questions vary in difficulty: if you get stuck on some part of a question, leave it and go on to the next one.

---

1. A graph G on n vertices is constructed using the following random process. Color each vertex with a color chosen independently and u.a.r. from a set of k colors. Finally, draw an edge between all pairs of vertices of the same color. The expected number of edges in G is exactly

$$\frac{n^2}{k^2} \qquad \frac{n}{k} \qquad \frac{n}{k^2} \qquad \frac{n^2}{k} \qquad \frac{\binom{n}{2}}{k}$$

---

2. Let G be a random graph in the $\mathcal{G}_{n,p}$ model.

(a) The expected number of 3-cliques in G is on the order of

$$np \qquad \frac{n^3}{p^3} \qquad np^3 \qquad n^3p^3 \qquad n^3p^6$$

(b) The threshold value of p for the existence of a 3-clique in G is on the order of

$$n^{-1/3} \qquad n^{-1} \qquad n^{-1/2} \qquad n^{-2/3} \qquad n^{-3/2}$$

---

3. **(a)** An unbiased coin is flipped repeatedly until <u>both</u> heads and tails are obtained. The expected number of times the coin is flipped is

$$2 \qquad \frac{5}{2} \qquad 3 \qquad \frac{10}{3} \qquad \frac{7}{2} \qquad \frac{9}{2}$$

(b) A fair 6-sided die is tossed repeatedly until <u>three</u> different outcomes are observed. The expected number of times the die is tossed is

$$2 \qquad \frac{11}{5} \qquad 3 \qquad \frac{33}{10} \qquad \frac{74}{20} \qquad 4$$

[continued overleaf]

4. **(a)** In a random skip list storing n elements, the expected height of any given element is

$O(1)$ $O(\log \log n)$ $O(\log n)$ $O(\sqrt{n})$ $O(n)$

(b) The expected maximum height of all the elements is

$O(1)$ $O(\log \log n)$ $O(\log n)$ $O(\sqrt{n})$ $O(n)$

---

5. A random variable X has expectation $\mathrm{E}(X) = 2$ and variance $\mathrm{Var}(X) = 9$, and the value of X never exceeds 10. Circle those <u>three</u> of the following statements that <u>must</u> be true about X:

$\Pr[X = 2] > 0$ $\Pr[X \geq 2] = \Pr[X \leq 2]$ $\mathrm{E}(X^2) = 13$

$\Pr[X \geq 6] \leq \frac{1}{3}$ $\Pr[X \geq 6] \leq \frac{9}{16}$ $\Pr[X \leq 1] \leq \frac{8}{9}$

---

6. A fair coin is tossed n times, where n is large. The probability of obtaining *exactly* $\frac{n}{2}$ heads is

$\sqrt{\frac{2}{\pi}}$ $\sqrt{\frac{2}{\pi}} \cdot \frac{1}{n}$ $\sqrt{\frac{2}{\pi n}}$ $\sqrt{\frac{2}{\pi}} \cdot \frac{\ln n}{n}$ $\sqrt{\frac{2}{\pi}} \cdot \frac{1}{2^n}$

---

7. Let π and π' be two independent, uniform random permutations of $\{1, \ldots, n\}$, and let π_0 be a *fixed* permutation. Which one of the following functions ρ from $\{1, \ldots, n\}$ to itself is <u>not</u> a uniform random permutation?

$\rho(x) = \pi(\pi'(x))$ $\rho(x) = \pi(\pi(x))$ $\rho(x) = \pi(\pi_0(x))$ $\rho(x) = \pi_0(\pi(x))$

---

8. **(a)** Suppose we are using the Schwartz-Zippel test to decide whether a polynomial $Q(x_1, \ldots, x_n)$ is identically zero. If Q has degree 100 and we want a success probability of at least $\frac{1}{2}$, the size of the set S from which we pick random values for the variables should be

2 100 200 400 2^{200}

(b) If $Q = (x_1 - a_1)(x_2 - a_2) \ldots (x_{100} - a_{100})$, for some constants $a_1, \ldots, a_{100}$, then with the above choice of $|S|$ the best lower bound we can claim on the success probability is

$\frac{1}{2}$ $\left(\frac{199}{200}\right)^{200}$ $\left(\frac{99}{100}\right)^{100}$ $\left(\frac{99}{100}\right)^{50}$ $\left(\frac{199}{200}\right)^{100}$

[continued overleaf]

9. **(a)** n balls are thrown into n bins. For large n, the probability that the first bin is empty is about

$$1-\frac{1}{\mathrm{e}} \qquad 1-\frac{1}{\mathrm{e}^2} \qquad \frac{1}{\mathrm{e}} \qquad \frac{1}{\mathrm{e}^2} \qquad \left(1-\frac{1}{\mathrm{e}}\right)^2 \qquad 1-\left(1-\frac{1}{\mathrm{e}}\right)^2$$

(b) n red balls and n blue balls are thrown into n bins. For large n, the probability that the first bin contains no red balls and three blue balls is about

$$\frac{3}{\mathrm{e}} \qquad \frac{1}{\mathrm{e}^4} \qquad \frac{1}{3\mathrm{e}^2} \qquad \frac{3}{\mathrm{e}^2} \qquad \frac{1}{\mathrm{e}^3} \qquad \frac{1}{6\mathrm{e}^2}$$

---

10. **(a)** Let a be a fixed n-bit vector such that $a \neq 0$, and let r be an n-bit vector chosen uniformly at random. The probability that the dot product $r \cdot a$ is equal to zero (mod 2) is

$$\frac{1}{2^n} \qquad \frac{1}{n} \qquad \frac{1}{4} \qquad \frac{1}{2} \qquad 1-\frac{1}{2^n}$$

(b) Now let R be an $n \times n$ 0-1 matrix chosen uniformly at random. The probability that the vector Ra has all components equal to zero (mod 2) is

$$\frac{1}{2^n} \qquad \frac{1}{n} \qquad \frac{1}{4} \qquad \frac{1}{2} \qquad 1-\frac{1}{2^n}$$

[continued overleaf]

11. Karger's algorithm

Recall Karger's algorithm for finding a minimum cut in an undirected graph G:

while G has more than two vertices **do**
    pick an edge $e = (u, v)$ u.a.r. and contract e
output the remaining edges

This question concerns the following graph G_n, consisting of a cycle of length $n - 1$ with a single edge attached to it:

(a) What is the minimum cut in G_n?

(b) Show that the probability that Karger's algorithm outputs the minimum cut in G_n is precisely $\frac{2}{n}$.

(c) The success probability in part (b) tends to zero as $n \to \infty$. Explain why this does not make Karger's algorithm useless in practice.

[continued overleaf]

12. Distributing knowledge

You are the CEO of a company employing n people. Your company is working on k different projects, each of which involves a group of at least r people; note that each individual employee may belong to several project groups.

Worried by their lack of basic skills, you decide to send your employees on a training course. There are two different courses offered (Course A and Course B), each covering a different set of skills. You decide that each employee should attend one course, but not both. However, it is also essential that in every project group at least one person goes to Course A and at least one person to Course B.

(a) Suppose you assign the employees independently and u.a.r. to the two courses. Show that the resulting assignment fails to satisfy your condition with probability at most $k2^{1-r}$.

(b) Part (a) implies that a valid assignment of employees to courses is possible provided the number of projects k is less than a certain value (which depends on r). What is this value? Justify your answer.

(c) Suppose now that $k \leq 2^{r-2}$. Design a randomized algorithm that produces a valid assignment with probability at least $\frac{1}{2}$. Justify the behavior of your algorithm.

[continued overleaf]

[**Q12 continued**]

(d) Consider now the situation in which there are *three* courses, A, B and C. As before, each employee can attend only one course, and in each project group at least one person must attend each course. Compute an upper bound analogous to that of part (a) for the probability that a random assignment fails to satisfy your condition in this case. How does this affect the value of k you computed in part (b)? [Hint: To figure out the probability that the assignment fails for some given group, apply inclusion-exclusion to the three events E_i = nobody in this group is assigned to course i.]

[**continued overleaf**]

13. Random walk on the cycle

This question concerns random walk on the n-vertex cycle C_n:

(a) Let i, j be any two vertices of C_n, and let d_{ij} be the distance (length of the shortest path) between them. What is the effective resistance R_{ij} between i and j, as a function of n and d_{ij}?

(b) What is the expected hitting time H_{ij} from i to j, as a function of n and d_{ij}? [Hint: Compute $H_{ij} + H_{ji}$ and then use symmetry.] For which pairs i, j is the hitting time largest (assuming for convenience that n is even)?

(c) Give an upper bound on the expected cover time of C_n, and show that it is tight up to a constant factor.

[continued overleaf]

14. How big is my database?

You are given the task of counting the number of records in a large database. The database is equipped with a "random sample" function, which returns a record chosen uniformly at random (and independently of all other samples). Here is a suggested algorithm:

repeat
  randomly sample a record
  if the record is unmarked **then** mark it
until you sample a record that is already marked
output X^2, where X = the number of records you marked

Your goal in this question is to demonstrate that, with reasonable probability, the output of this algorithm lies within a factor of 10 of the true number of records N; i.e., that $\Pr[\frac{N}{10} \leq X^2 \leq 10N]$ is quite large.

(a) Recall the *birthday problem*, which says that in a world with n days per year, the number of people we need to invite to a party in order to have a reasonable chance of two people sharing the same birthday is some constant times $\sqrt{n}$. Using this observation, give a *brief, informal* explanation of why the above algorithm might give a reasonable estimate of N. [Note: you are **not** required to prove anything here.]

(b) Now let's analyze the algorithm in detail. As in the algorithm, let the r.v. X be the number of records sampled before we get the first repeat. Show that

$$\Pr[X > k] = \left(1 - \frac{1}{N}\right)\left(1 - \frac{2}{N}\right)\left(1 - \frac{3}{N}\right)\cdots\left(1 - \frac{k}{N}\right).$$

(c) Use the approximation $e^{-2x} \leq 1 - x \leq e^{-x}$ (which is valid for $|x| \leq \frac{1}{2}$) to deduce that, for all $k \leq \frac{N}{2}$,

$$e^{-2\sum_{i=1}^{k} i/N} \leq \Pr[X > k] \leq e^{-\sum_{i=1}^{k} i/N}.$$

[continued overleaf]

[**Q14 continued**]

(d) Use the fact that $\sum_{i=1}^{k} i$ lies between $\frac{k^2}{2}$ and k^2 to deduce that, for all $k \leq \frac{N}{2}$,

$$e^{-2k^2/N} \leq \Pr[X > k] \leq e^{-k^2/2N}.$$

(e) Finally, by substituting two suitable values for k in the bounds of part (d), deduce that the probability that the value X^2 output by the algorithm falls <u>outside</u> the range $[\frac{N}{10}, 10N]$ is at most

$$(1 - e^{-1/5}) + e^{-5} \approx 0.19.$$

[Note: You may assume that the bounds in part (d) hold even when k is not an integer.]

[**continued overleaf**]

15. Randomized algorithms with two-sided errors

Let Π be some problem with yes/no answers on every input. We will denote by $\Pi(x)$ the answer to Π on input x. This question is concerned with reducing the error probability of randomized algorithms with *two-sided* errors.

(a) We begin with the familiar case of one-sided errors. Suppose Algorithm $\mathcal{A}$ behaves as follows: on every input x,

(i) if $\Pi(x) =$ yes then $\mathcal{A}$ outputs "yes" with probability $\geq \frac{1}{2}$, and "no" otherwise;

(ii) if $\Pi(x) =$ no then $\mathcal{A}$ outputs "no" with probability 1.

Explain how to modify $\mathcal{A}$ so that the probability of error in case (i) is at most ϵ, for any desired $\epsilon > 0$, and the increase in running time is a factor of $O(\log(\frac{1}{\epsilon}))$.

(b) In preparation for dealing with two-sided errors, we first derive a useful fact about coin-tossing. Recall the Chernoff bound: if X is the number of heads in n independent tosses of a coin with heads probability p, then $\Pr[X \leq (1-\delta)\mu] \leq e^{-\delta^2\mu/2}$, where $\mu = np$. Use this bound to prove that, if X is the number of heads in n tosses of a coin with heads probability $\frac{3}{4}$,

$$\Pr[X \leq \tfrac{n}{2}] \leq e^{-n/24}.$$

(c) Now consider Algorithm $\mathcal{B}$, which has two-sided error and behaves as follows: on every input x,

(i) if $\Pi(x) =$ yes then $\mathcal{B}$ outputs "yes" with probability $\geq \frac{3}{4}$, and "no" otherwise;

(ii) if $\Pi(x) =$ no then $\mathcal{B}$ outputs "no" with probability $\geq \frac{3}{4}$, and "yes" otherwise.

Show how to modify $\mathcal{B}$ so that the probability of error in both cases is at most ϵ, for any desired $\epsilon > 0$, and the increase in running time is a factor of $O(\log(\frac{1}{\epsilon}))$. [Hint: Use part (b).]

[The End]