CS49/Math59 Lab 5

This lab assignment is due **before the start of class** on Wednesday, November 4. Homework handed in during class but after I begin the lecture will be counted as late submissions. Some things to note:

- This is a **two week lab**.
- I encourage you to write your solution using LaTeX, but you are not required to.
- You may have one partner for this assignment, but are not required to. If you work with a partner, submit just one writeup.
- Aside from your partner, you should not discuss problems in detail with anyone. It's OK to discuss approaches at a high level. In fact, I encourage you to discuss general strategies. However, you should not reveal specific details of a solution, nor should you show your written solution to anyone else.

Make sure your names are on your submission, and show your work to maximize partial credit.

1. **Analysis.** When doing research (in theoretical computer science or another discipline) it's a good idea to think critically about the existing research and question whether previous researchers could have done things differently and/or better. This has several benefits—it deepens your understanding of existing results, it makes it easier to apply new ideas to achieve new results, and occasionally you find that existing results aren't optimal and get new results (and publications) as a consequence.

   In this problem, you will examine some of the recent probabilstic method results you've seen in class and consider whether the choices we've made have been the best possible choices. Perhaps with a tweak of the techniqeus we've seen, we can strengthen the results.

   (a) In class, we saw a result that used alterations to show that any graph with n vertices and $nd/2$ edges has an independent set of size at least $n/(2d)$. We started by adding each vertex independently to S with probability p, and than removing a vertex for each edge (i, j) with both endpoints in S. The resulting set was independent and had expected size
   $$E[|S|] \geq np - \frac{nd}{2}p^2 \ .$$
   Choosing $p = 1/d$ gave an independent set of size at least $n/2d$. Show this is the best we can do using the alteration procedure from class by maximizing
   $$f(p) = np - \frac{nd}{2}p^2 \ .$$

   (b) To find a set S of n points in the unit square that had $T(S) = \Omega(1/n^2)$, we started by picking $2n$ points uniformly, and then removed on point of each triangle that had area $\leq 1/100n^2$. We saw that the expected number of such "small-area" triangles was at most n, so there must exist a choice of $2n$ initial points such that the process returns a set of

n points with no small-area triangles. In this way, we proved $T(n) \geq T(S) \geq 1/100n^2$, hence $T(n) = \Omega(1/100n^2)$.

Is this the best we can do? Can we get a stronger lower bound for $T(n)$ by initially picking $3n$ (or $1.5n$ or $10n$ or ...) points? What happens if we eliminate a point from each triangle with area less than $1/50n^2$ or $100/n^2$ or even $1/100n^{1.5}$? Using the alteration technique, try to improve the bound of $T(n) \geq 1/100n^2$. Either give an improvement, or some analysis showing such an improvement is unlikely without new ideas.

(c) Maximize $f(x) = n^2 x - \frac{n^6}{9} x^9$. You may assume that n is greater than zero and x is nonnegative.

2. **Probability.** (Shoup exercise 8.26) For real-valued random variables X and Y, the *covariance* is defined as $\mathrm{Cov}[X, Y] := E[XY] - E[X]E[Y]$. Show that:

(a) If X, Y are independent then $\mathrm{Cov}[X, Y] = 0$.

(b) If X, Y, Z are real-valued random variables and a is a real number, then $\mathrm{Cov}[X+Y, Z] = \mathrm{Cov}[X, Z] + \mathrm{Cov}[Y, Z]$ and $\mathrm{Cov}[aX, Z] = a\,\mathrm{Cov}[X, Z]$.

(c) If $X_1, \ldots, X_n$ are real-valued random variables, then

$$\mathrm{Var}[\sum_i X_i] = \sum_{i=1}^{n} \mathrm{Var}[X_i] + \sum_{i \neq j} \mathrm{Cov}[X_i, X_j] \ .$$

3. **Lucky Coin.** You and your friend Rich often eat lunch in Media rather than sampling some of the fine meals that Sharples offers. (Of course, you never skip Bagel Bar). Each time you eat lunch, you and Rich flip a coin to see who gets to pay. Rich always uses his "lucky" silver dollar, and lately, you've been having a funny feeling that Rich is paying less often than he should. Could it be that your so-called friend is cheating you?

In this problem, you will design and analyze an algorithm to detect if a coin is fair or biased. Assume that Rich's silver dollar is either a fair coin, or it has a probabilty of 0.6 of being heads. Your algorithm may make mistakes, but should have error probability of at most 2^{-30}. (Rich is sensitive, and you want to make very very sure that he is cheating before you accuse him). Use as few coin flips as possible. How many coin flips does your algorithm use?

4. **Magic Square.** This problem generalizes the Magic Square exercises you saw in lab. Let $[n] := \{1, \ldots, n\}$. For any integers n, m, d with $n/2 < m < n$, let $U := [n]^d$ be set of all d-dimensional vectors, each entry of which comes from $[n]$. For example, $(1, 5, 3, 2) \in \{1, 2, 3, 4, 5\}^4$. Define a *rectangle* to be a subset of U if $R = R_1 \times \cdots \times R_d$ where $R_i \subseteq [n]$ and $|R_i| = m$. (Think of $R_1, \cdots, R_d$ as selecting m out of n possible elements for each coordinate)

Let $\mathcal{S}$ be the set of all possible rectangles of U. Say that a rectangle R *covers* p if $p \in R$. Finally, say that a set of rectangles $\mathcal{C} \subseteq \mathcal{S}$ *covers* U if every point[1] $p \in U$ is *covered* by some $R \in \mathcal{C}$. We'd like to minimize the number of rectangles needed to cover U.

For example, we saw in lab that 3 two-by-two rectangles are needed to cover $[3]^2$, and that 5 two-by-two-by-two rectangles are needed to cover $[3]^3$.

Fix integers n, m, d such that $n/2 < m < n$.

[1] I will often use "point" to refer to an arbitrary element of U

(a) How many points are covered by an rectangle? How many points are there in U?

(b) Use your answers from (a) to determine a simple *lower bound* on the number of rectangles needed to cover U.

(c) fix an arbitrary $p \in U$ and choose a rectangle uniformly at random. What is the probabiltiy that R covers p?

(d) Use the probabilistic method to derive an *upper bound* on the number of rectangles needed to cover U.

(e) Now, suppose you have the following communication game. Alice and Bob agree in advance on a set of rectangles that cover U. Alice has a point $p \in U$ and wants to communicate to Bob a rectangle containing p as efficiently as possible. (For example, we covered $[3]^2$ with three two-by-two rectangles in class. Given this covering and any $p \in U$, Alice could communicate which rectangle contained her point using $\lceil \log_2 3 \rceil$ bits.) Using your upper and lower bounds on the number of rectangles needed to cover U, give an estimate of how many bits of communication are required.

5. **Square-free matrices.** Call an $n \times n$ binary matrix M *square-free* if there is no 3×3 submatrix with all 1 entries. For example, the following 4×4 matrix is square-free

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

whereas the following 4×4 matrix is *not* square-free because the each entry is a 1 in the submatrix $\{1, 3, 4\} \times \{1, 2, 4\}$:

$$\begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix}$$

There are applications in theoretical computer science where we'd like binary matrices to be square-free and have as many one-entries as possible. Let $Z(n)$ be the maximum number of one entries in a square-free $n \times n$ binary matrix.

Use the alteration technique to give a lower bound on $Z(n)$ — your answer should state that "there exists a square-free binary matrix with X one entries". Express X as a function of n and make it as large as you can.

6. **Attribution.** Did you get assistance on any of the problems on this assignment from anyone aside from me and/or your lab partner? For example, did you discuss any problems at a high level with other students? Did you accidentally stumble on solutions while doing a websearch on related material? If so, descibe the nature of the assistance here. (e.g. "We briefly discussed problem 1 with X,Y, and Z" or "We saw a solution on ⟨this website⟩ before finding our own solution") If you (and your partner) worked alone, please say so here.

7. **Lab Questionnaire.** (None of these questions will have an impact on your grade, this is to help provide the feedback I need to make the course the best it can be)

(a) Approximately how many hours per partner did you spend on this lab?

(b) How difficult did you find this lab? (enter a number 1-5, with 5 being very difficult and 1 being very easy)

(c) Describe the biggest challenge you faced on this lab.