MPRI 2.3.2, Foundations of Privacy
Final exam

The exam consists of several questions, each of them contributing for a certain number of points, indicated between parentheses. The maximum total score is 20.

Question 1 (3 points)

Randomized response is a technique developed in the social sciences to collect statistical information about embarassing or illegal behavior. Assume, for instance, that the study is about "cheating on exams" ☺. Study participants are told to report whether or not they have ever cheated on an exam, according to the following algorithm:

(i) Flip a fair coin.

(ii) If the result is *heads*, then respond truthfully.

(iii) If the result is *tails*, then flip the coin a second time, and respond "Yes" if the result is *heads* and "No" if it is *tails*.

The idea is to motivate people to participate in the study and to respond truthfully. In fact a "Yes" answer is not incriminating, since this answer occurs with probability at least $\frac{1}{4}$ whether or not the respondent has actually cheated. In the following, we assume that everybody adheres to the algorithm entirely (i.e., that everybody responds truthfully in case (ii)).

1.1 Say whether or not the mechanism is ε-differentially private. In the affirmative case calculate ε. In the negative case, find a way to add noise so to obtain a differentially-private mechanism.

1.2 The purpose of the study, naturally, is to approximate as much as possible the real percentage of "Yes" vs. "No". Assuming that we have no prior bias on such percentage (i.e., all percentages are equally probable *a priori*), find the optimal remapping. Namely, if the percentage of reported "Yes" is p, what is the best guess for the real percentage?

Question 2 (5 points)

Consider a medical database which contains records people affected by one or more of 4 diseases, A, B, C, and D. Assume that we want to study what is the most common disease, i.e. we are interested in the query "What is the disease that affects the most people in the database?". We want to define an ε-differentially private mechanism to answer this query, with a utility as high as possible. The natural notion of utility, here, is the minimization of the difference between the number of people affected by the disease y and the number of people affected by the disease y', where y is the real answer and y' is the reported answer.

One obvious mechanism would be to report the real answer with probability p, and any other possible answer with probability $pe^{-\varepsilon}$ (where p is a normalization constant). However, there is a mechanism that gives a better utility for the same level ε of privacy. Find such mechanism, prove that it is ε-differentially private, and justify the improvement in utility.

Hint: use a combination of counting queries (one for each of the diseases A, B, C, and D), add noise to them, and then compute the max.

Question 3 (2 points)

Geo-indistinguishability is compositional, in the same sense as differential privacy. Namely, if we have a trace (a sequence of locations, typically correlated to each other) and we apply a geo-indistinguishable mechanism, independently, to each location in this sequence, then the reported sequence of location has the geo-indistinguishability property. State this property formally, i.e. with a precise formula, and prove it. (Give such formula in the most general form, i.e., the mechanisms applied in different locations may have different levels of geo-indistinguishability.)

Question 4 (3 points)

The following is known as the "Three Prisoners problem". Three prisoners, A, B and C are sentenced to death, but one of them (uniformly chosen at random) is selected to be pardoned (so 2 out of 3 prisoners will be executed). The warden knows which one will be pardoned but is not allowed to tell the prisoners.

Prisoner A begs the warden to let him now the identity of *one of the others* who will be *executed*: "if B is pardoned, give me C's name and vice versa. If I'm pardoned, choose randomly to name B or C".

4.1 Model the problem as a channel and compute its multiplicative Bayes-capacity. Using the capacity, compute the probability of correctly guessing the pardoned prisoner after receiving the warden's answer (properly justify why the capacity is relevant for this task).

4.2 Prisoner A is of course only interested is finding information *about himself*, i.e. whether *he* is going to be executed or not. Is the warden's answer useful for A? Justify your answer by an Information Flow analysis of a properly constructed channel (either the same channel of Question 4.1 or a different one).

Question 5 (3 points)

5.1 Let N be a non-interfering channel, let R be an arbitrary channel and let $C = NR$. Show that C is also non-interfering, using Information Flow arguments.

5.2 Let A be a channel with two linearly dependent columns y_1, y_2, i.e. there exist $k \in \mathbb{R}$ such that $A_{xy_1} = kA_{xy_2}$ for all $x \in \mathcal{X}$.

Let B be the channel obtained from A by replacing the columns y_1, y_2 with their sum, i.e. a single column y such that $B_{xy} = A_{xy_1} + A_{xy_2}$.

Show that A and B have exactly the same leakage (for all prior and gain functions) in two ways: (i) by considering the hyper-distributions produced by each channel and (ii) using composition refinement.

Question 6 (4 points)

6.1 Consider an instance of the Dining Cryptographers protocol on an *arbitrary connection graph* (meaning there is an arbitrary number of users, and each user is connected to an arbitrary subset of other users). As usual, each connected pair of users share a coin, but coins are not assumed to be fair, each coin has its own probability of giving 0 or 1. This system might or might not leak information, depending on the connecting graph and the probabilities of the coins.

Now on this Dining Cryptographers instance, we *add a new coin*, meaning that we connect two previously disconnected users who now share a coin. The new coin might also be biased.

Show that the new protocol instance cannot leak more information than the first one.

6.2 Use the above result to show that a Dining Cryptographers instance with 3 users and 3 coins on a ring graph, where one of the coins is biased, and the remaining 2 coins are fair, has no leakage.

What about the same ring (3 cryptographers and 3 coins), where 2 coins are biased, and one is fair?

As always, properly justify all answers.