1. Multiplicative secret-sharing. We saw that both the \( t \)-private CNF scheme and the \( t \)-private Shamir scheme are multiplicative whenever \( n > 2t \). Is this also true for the DNF scheme? Prove your answer.

2. Security vs. privacy. Intuitively, security (against an active adversary) is stronger than privacy (against a passive adversary). Prove or disprove the following statement: if a protocol \( \pi \) is a perfectly \( t \)-secure protocol for \( f \) then it is also a perfectly \( t \)-private protocol for \( f \).

3. On composing private protocols. Let \( f \) be a deterministic functionality. We say that a protocol \( \pi \) privately computes \( f \) with respect to an active adversary if: (1) \( \pi \) correctly computes \( f \) when no party is corrupted; and (2) The view of an active adversary alone (i.e., without concatenating the outputs of uncorrupted parties) can be simulated in the ideal model. (Other than omitting the outputs of uncorrupted parties the definition is identical to the usual definition of security.)

Prove or disprove: under the notion of perfect \( t \)-privacy with respect to an active adversary, if \( \pi^f \mid g \) is a private reduction from \( f \) to \( g \) and \( \pi^g \) is a private protocol for \( g \) then the protocol \( \pi^f \) obtained from \( \pi^f \mid g \) by implementing the oracle calls using \( \pi^g \) is a private protocol for \( f \).

4. Secure computation in two rounds.

   (a) A degree-\( d \) (multivariate) polynomial is a sum of monomials, where each monomial is the product of at most \( d \) variables. Show that two rounds are sufficient for \( n \) parties to \( t \)-privately evaluate an arbitrary degree-\( d \) polynomial \( p \) of their inputs (over some finite field \( F \)), as long as \( n > dt \). Explain why the protocol is private (a formal proof is not required).

   **Hint:** First generalize the MS functionality to handle \( d \) secrets, and use it to obtain a 3-round protocol. Then save one round.

   (b) Conclude that any \( n \)-party functionality has a 2-round perfectly \( t \)-private protocol if \( n > 3t \). (Here the protocol is not required to be efficient.)

   (c) Describe a 2-round perfectly 1-secure coin flipping protocol for \( n = 5 \) parties. Explain why the protocol is secure.

   Note: this part of the question is unrelated to the two previous parts.

5. Completeness in secure computation.

   (a) A \( k \)-party functionality \( g \) is said to be complete (for private computation) if there is a perfectly \((n - 1)\)-private reduction from any \( n \)-party functionality \( f \) to \( g \). We saw in class that the oblivious transfer (OT) functionality is complete. For each of the following functionalities, determine whether it is complete. Prove your answers.

      i. \( g(x_1, x_2) = (x_1 \lor x_2, \bot) \)
      ii. \( g(x_1, (x_2, x_2')) = (x_1 \land x_2) \oplus x_2' \) (here \( g \) has two inputs and a single public output).
      iii. \( g(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus x_3 \)

   (b) Consider the notion of completeness in the model of perfect security against an active adversary. An \( n \)-party functionality \( g \) is called trivially complete if any \( n \)-party functionality \( f \) can be \((n-1)\)-securely reduced to \( g \) by means of a single call to \( f \), without any direct communication between
the parties. We may assume without loss of generality that \( f \) is restricted to be a deterministic, single-output functionality. Ignoring computational efficiency issues, prove that there exists such a trivially complete functionality \( g \).

**Hint:** The \( i \)th input to \( g \) contains a description of a function \( f_i \) along with an input \( x_i \) to \( f_i \). First consider what happens if the \( i \)th output of \( g \) is defined as \( f_i(x_1, \ldots, x_n) \). Then consider a better idea.

6. Communication-efficient oblivious transfer using homomorphic encryption. A homomorphic encryption scheme (over \( \mathbb{Z}_2 \)) is a randomizable encryption scheme \((G, E, D)\) (as defined in class) satisfying the following additional property. There exists an efficient “combiner” algorithm \( H \), such that given an arbitrary pair of ciphertexts \( c_1 \in E(pk, b_1) \) and \( c_2 \in E(pk, b_2) \), \( H(pk, c_1, c_2) \) returns a ciphertext \( c \in E(pk, b_1 + b_2) \) (where addition is in \( \mathbb{Z}_2 \)).

(a) Show that the GM encryption scheme described in class is homomorphic.

(b) From here on, you may assume that homomorphic encryption exists. Define the following generalization of the OT functionality: \( \binom{k}{1} \)-OT\((\{a_1, \ldots, a_k\}, b) = (\perp, a_b) \), where \( a_i \in \{0, 1\} \) and \( b \in [k] \).

Describe a 2-round computationally private protocol for \( \binom{k}{1} \)-OT in which \( P_2 \) sends \( k \) ciphertexts to \( P_1 \) and receives a single ciphertext in return.

(c) A private information retrieval (PIR) protocol allows a user to retrieve a data item from a database while hiding the identity of her selection from the server storing the database. PIR can be formally defined as a computationally \( \mathcal{T} \)-private protocol for the \( \binom{k}{1} \)-OT functionality, where \( \mathcal{T} = \{\emptyset, \{P_1\}\} \). (Here \( P_1 \) is the server and \( P_2 \) is the user.) Note that PIR is weaker than OT in that it is only concerned with the user's privacy and does not prevent the user from learning additional information.

Describe a PIR protocol whose communication complexity (the number of bits exchanged) is sublinear in \( k \).

**Hints:** (1) View the database as a \( \sqrt{k} \times \sqrt{k} \) matrix \( A \) and allow the user to retrieve an entire column of \( A \); (2) choose the security parameter for the encryption scheme so that the length of each ciphertext is made sufficiently small.

(d) Show how to modify the previous protocol so that it realizes \( \binom{k}{1} \)-OT with sublinear communication.