Problem 1

In this question you will explore the difference between secure and strongly secure MACs for fixed message length. Recall that the two differ in the type of forgery: A forgery is a pair (M, T) such that $Ver(K, M, T) = 1$ and M was never queried from the tagging oracle, while a weak forgery is a pair (M, T) such that the query-answer pair (M, T) was never observed in the interaction between the adversary and the tagging oracle. (So in a weak forgery, M could have been queried before, but if the oracle did not return T as an answer when M was queried, (M, T) is considered a weak forgery.) A MAC that prevents forgeries is called secure (against chosen message attack); one that prevents weak forgeries we will call strongly secure.

(a) Assuming that secure MACs for message length m exist, show that there exists a MAC for message length m that are secure but not strongly secure.

(b) Show that if Tag is deterministic and (Tag, Ver) is secure, then there exists a verifier Ver' such that (Tag, Ver') is strongly secure.

Problem 2

Consider the following encryption scheme, where $\{F_K\}$ is a pseudorandom function family:

$$Enc((K_1, K_2), M) = (S, F_{K_1}(S) + M, F_{K_2}(S + M)) \quad \text{where } S \text{ is a random string}$$

$$Dec((K_1, K_2), (S, C, T)) = \begin{cases} C + F_{K_1}(S), & \text{if } F_{K_2}(S + C + F_{K_1}(S)) = T \\ \texttt{error}, & \text{otherwise.} \end{cases}$$

(a) Show that if (Enc, Dec) is used with the same key $K_1 = K_2$, the scheme is not message indistinguishable, even for one encryption.

(b) Now assume that K_1 and K_2 are independent. Consider the ideal scheme $(REnc, RDec)$ which is a variant of (Enc, Dec) where F_{K_1} and F_{K_2} are replaced with truly random independent functions R_1 and R_2, respectively. Show that if $\{F_K\}$ is pseudorandom and $(REnc, RDec)$ is CPA-secure, than (Enc, Dec) is CPA secure (for an appropriate choice of the parameters).

(c) Show that $(REnc, RDec)$ is CPA-secure (for an appropriate choice of the parameters).

(d) **(Optional)** Can you show that (Enc, Dec) is CCA-secure?

Problem 3

Assuming cryptographic hash families exist, show that there is a function family $\{h_K \colon \{0,1\}^{2k} \to \{0,1\}^k\}$ which is a cryptographic hash family but is not a pseudorandom function family.

Problem 4

Suppose Alice has some database that Bob wants to query interactively. Bob wants to be sure that Alice's answers are consistent: they are independent of his queries and the order in which he asks them. One solution is for Alice to first commit to her database, and for Bob to verify that Alice's answers are consistent with her commitment.

We will think of a database as a function $F \colon \{0,1\}^q \to \{0,1\}^k$ that takes a query in $\{0,1\}^q$ and outputs an answer in $\{0,1\}^k$. The size of a database is the size of the truth table of F, that is $2^q k$. You should think of the database as small enough so that Alice can perform computations on the whole database, but large enough so that it is infeasible for Alice to send Bob the full database.

Assume Alice and Bob share a random key $K \in \{0,1\}^k$. A *database commitment scheme* consists of three algorithms (Com, Ans, Ver) where

- $Com(K, F)$ is a *commitment algorithm* that takes a key K and a database F and outputs a string in $\{0,1\}^k$, which is a commitment to F.
- $Ans(K, C, x)$ is an *answering algorithm* that takes a key K, a commitment C to a database F, and a query $x \in \{0,1\}^t$ to F and outputs $F(x)$.
- $Ver(K, C, x, a)$ is a *verification algorithm* that takes a key K, a commitment C, a query x and an answer a, accepts if the query-answer pair is consistent with the commitment, and rejects otherwise.

We will say that the database commitment scheme is secure if it is infeasible for a computationally bounded adversary to commit to a database F, but answer queries according to some other database F' without getting caught.

(a) Give a formal definition of an (s, ε)-secure database commitment scheme. You should give separate definitions for functionality and for security.

(b) Let $\{h_S \colon \{0,1\}^{2k} \to \{0,1\}^k\}$ be a cryptographic hash family for input length $2k$. Let $h_S^{(q)} \colon \{0,1\}^{2^t k} \to \{0,1\}^k$ be the function recursively defined by the formula

$$h_S^{(q)}(M_1 M_2) = h_S(h_S^{(q-1)}(M_1), h_S^{(q-1)}(M_2)), \quad M_1, M_2 \in \{0,1\}^{2^{q-1}k}$$

with $h_S^{(1)} = h_S$. Show that $\{h_S^{(t)}\}$ is a cryptographic hash family for input length $2^q k$ (for appropriate parameters).

(c) Use part (b) to construct a secure database commitment scheme and prove your scheme is secure assuming $\{h_S\}$ is a secure cryptographic hash family.