Problem 1.

The answer is (c).

MAC' is not a secure MAC. If the adversary repeatedly query the tag of some messages m, and gets tags $(r, \tau_1, \tau_2), (r', \tau_1', \tau_2')$. Then (r, τ_1, τ_2') is a valid tag for $m \oplus r \oplus r'$.

Problem 2.

Part A. Encoder E is an efficient algorithm. Decoder D simply removes all leading 0's and the first 1.

For any $x \neq y$, we show that $E(x)$ is not a prefix of $E(y)$:

- If $|x| > |y|$, then $E(x)$ is not a prefix of $E(y)$ because $E(x)$ is longer.
- If $|x| = |y|$, then $E(x)$ is not a prefix of $E(y)$ because $|E(x)| = |E(y)|$ and $E(x) \neq E(y)$.
- If $|x| < |y|$, then $E(x)$ is not a prefix of $E(y)$ because the $(|x|+1)$-th bit of $E(x)$ is 1 and the $(|x|+1)$-th bit of $E(x)$ is 0.

Part B. A simple encoding satisfying the requirements is $E(x) = 0^{|\ell|}1\|\ell\|x$, where ℓ is the bit representation of $|x|$.

More generally, given any prefix-free encoding E, we can construct another prefix-free encoding $E'(x) = E(\ell)\|x$.

Part C. There does not exist a prefix-free encoding E such that $|E(x)| = |x| + o(\log|x|)$. This can be proved by contradiction. Assume such encoding E exists, then there is a number $N \in \mathbb{N}$ such that $\forall x \in \{0,1\}^*, |x| > N \implies |E(x)| \leq |x| + \log|x|$. (A common mistake is missing N.)

Consider a random infinite-length binary string R. For any bit string $s \in \{0,1\}^*$, the probability s is a prefix of R is $2^{-|s|}$. Because E is a prefix-free encoding, the events "$E(x)$ is a prefix of R" and "$E(x')$ is a prefix of R" are disjoint, for any distinct x, x'. So

$$1 \geq \sum_{x\in\{0,1\}^*} \Pr[E(x) \text{ is a prefix of } R] = \sum_{x\in\{0,1\}^*} 2^{-|E(x)|}.$$

This contradicts our assumption, since $\forall x \in \{0,1\}^*, |x| > N \implies |E(x)| \leq |x| + \log|x|$ implies

$$\sum_{x\in\{0,1\}^*} 2^{-|E(x)|} \geq \sum_{n>N} \sum_{x\in\{0,1\}^n} 2^{-|E(x)|} \geq \sum_{n>N} 2^n 2^{-n-\log n} = \sum_{n>N} \frac{1}{n} = +\infty$$

Part D. Let $\ell = |x| < 2^\lambda$. The first λ-bit block of $E(x)$ encodes ℓ. Then append x to the encoding. Finally appends (at most $\lambda - 1$) 0s to the encoding so that the length is a multiple of λ.

The length of the encoding is less then $|x| + 2\lambda$, and one can easily verify it is a prefix-free encoding.

Problem 3.

Part A. First, we introduce a hybrid world which is similar to the real world, except PRF $F(k, \cdot)$ is replaced by a truly random function $f : \{0,1\}^\lambda \to \{0,1\}^\lambda$. Since F is a PRF, the hybrid world is indistinguishable from the real world.

It remains to prove that $\tilde{F}_{\text{CBC}}$ (defined as follows) is indistinguishable from a random function under prefix-free querying.

$$\begin{aligned}\tilde{F}_{\text{CBC}}(m_1, m_2, \dots, m_\ell) &:= \begin{cases} \tilde{f}(m_\ell \oplus \tilde{F}_{\text{CBC}}(m_1, m_2, \dots, m_{\ell-1})), & \text{if } \ell > 1 \\ f(m_1), & \text{if } \ell = 1 \end{cases} \\ &= f(m_\ell \oplus f(m_{\ell-1} \oplus \dots f(m_2 \oplus f(m_1)) \dots)).\end{aligned}$$

For any message $M = (m_1, m_2, \dots, m_\ell)$. Define

$$\begin{aligned}\mathcal{C}(M) &= \big(m_1, \tilde{F}_{\text{CBC}}(m_1) \oplus m_2, \dots, \tilde{F}_{\text{CBC}}(m_1, m_2, \dots, m_{\ell-1}) \oplus m_\ell\big), \\ \mathcal{C}_{\text{tail}}(M) &= \tilde{F}_{\text{CBC}}(m_1, m_2, \dots, m_{\ell-1}) \oplus m_\ell,\end{aligned}$$

i.e., the ℓ calls and the last call to f when computing $\tilde{F}_{\text{CBC}}(M)$.

Let $M_1, M_2, \dots, M_t$ be the first t queries made by the adversary, define

$$\begin{aligned}\mathcal{C}(M_1, \dots, M_t) &= \mathcal{C}(M_1) \cup \dots \cup \mathcal{C}(M_t), \\ \mathcal{C}_{\text{tail}}(M_1, \dots, M_t) &= \{\mathcal{C}_{\text{tail}}(M_1), \dots, \mathcal{C}(M_t)\}.\end{aligned}$$

We expect the following statements to hold,

(A_t) There is no collision among the first t queries with overwhelming probability.

Given two messages $M = (m_1, \dots, m_\ell)$, $M' = (m'_1, \dots, m'_{\ell'})$, let $\mathcal{C}(M) = (t_1, \dots, t_\ell)$, $\mathcal{C}(M') = (t'_1, \dots, t'_{\ell'})$. We say there is a collision between M, M' if $t_i = t'_j$ and $(m_1, \dots, m_i) \neq (m'_1, \dots, m'_j)$ for some i, j.

We say there is no collision among the first t queries if there is no collision between M, M' for any $M, M' \in \{M_1, \dots, M_t\}$.

(B_t) Intermediate outputs of f are hidden from the adversary: the joint distribution of $f(x)$ for any $x \in \mathcal{C}(M_1, \dots, M_t) \setminus \mathcal{C}_{\text{tail}}(M_1, \dots, M_t)$ is close to uniform conditioning on the adversary's view.

(C_{t+1}) The distribution of $\tilde{F}_{\text{CBC}}(M_{t+1})$ is close to uniform conditioning on the adversary's view after the first t queries.

The statements can be proved by induction.

($A_t \implies B_t$) If there is no collision, the outputs of $f(x)$ for all $x \in \mathcal{C}(M_1, \dots, M_t)$ are i.i.d. uniform. The adversary only learns $f(x)$ for all $x \in \mathcal{C}_{\text{tail}}(M_1, \dots, M_t)$, which reveals no information about the rest of $f(x)$.

($A_t + B_t \implies A_{t+1}$) Let $M = (m_1, \dots, m_\ell)$ be the $(t+1)$-th query. Define i be the largest index such that $(m_1, \dots, m_i)$ is the prefix of a previous message. By the prefix-free requirement, $(m_1, \dots, m_i)$ does not equals M (i.e., $i < \ell$) or any previous message. Thus, together with the non-collision statement A_t, $\tilde{F}_{\text{CBC}}(m_1, \dots, m_{i-1}) \oplus m_i \notin \mathcal{C}_{\text{tail}}(M_1, \dots, M_t)$. By the statement B_t, the distribution of $\tilde{F}_{\text{CBC}}(m_1, \dots, m_i)$ is close to uniform conditioning on the adversary after the first t queries. Thus despite the

Problem 3 continued on next page...

adversary's strategy of choosing m_{i+1}, $\tilde{F}_{\text{CBC}}(m_1, \ldots, m_i) \oplus m_{i+1} \notin \mathcal{M}(M_1, \ldots, M_t)$ with overwhelming probability. Then we argue inductively for each $i < j \leq \ell$,

$$\tilde{F}_{\text{CBC}}(m_1, \ldots, m_j) = f(\tilde{F}_{\text{CBC}}(m_1, \ldots, m_{j-1}) \oplus m_j)$$

is a fresh sample, which does not collide with any previous value with overwhelming probability, and f is not called on $\tilde{F}_{\text{CBC}}(m_1, \ldots, m_j) \oplus m_j$ with overwhelming probability.

($A_{t+1} \implies C_{t+1}$) Let $M = (m_1, \ldots, m_\ell)$ be the $t+1$-th query. A_{t+1} and the prefix-free requirement imply that $\tilde{F}_{\text{CBC}}(m_1, \ldots, m_{\ell-1}) \oplus m_\ell$ is not in $\mathcal{C}(M_1, \ldots, M_t)$. Thus the output $\tilde{F}_{\text{CBC}}(M_{t+1})$ is a fresh sample even conditioning on all the adversary's knowledge.

Part B. Since E is a prefix-free encoding, the MAC scheme only calls $F_{\text{CBC}}(k, \cdot)$ on inputs among which no one is the prefix of another. Therefore, $F_{\text{CBC}}(k, \cdot)$ can be replaced by a random function $f : \{0,1\}^* \to \{0,1\}^\lambda$. Formally, define another MAC scheme $\widetilde{\Pi} = (\widetilde{\mathsf{Gen}}, \widetilde{\mathsf{MAC}}, \widetilde{\mathsf{Verify}})$ as

- $\widetilde{\mathsf{Gen}}$ samples a random function f.
- $\widetilde{\mathsf{MAC}}(f, m) = f(E(m))$.
- $\widetilde{\mathsf{Verify}}$ is canonical.

Then

$$\left|\Pr[\mathsf{Macsforge}_{\mathcal{A},\Pi}(1^\lambda) = 1] - \Pr[\mathsf{Macsforge}_{\mathcal{A},\widetilde{\Pi}}(1^\lambda) = 1]\right| \leq \text{negl}(\lambda)$$

because F_{CBC} is a prefix-free PRF.

To completes the proof, we argue that $\Pr[\mathsf{Macsforge}_{\mathcal{A},\widetilde{\Pi}}(1^\lambda) = 1]$ is negligible. To forge the tag of a message m, the adversary has guess $f(E(m))$ correctly, the probability is $\frac{1}{2^\lambda}$ since f is a random function.

Problem 4. *Answer provided by George Ma*

The scheme does not satisfy unforgeability, since any λ-bit string is a valid ciphertext.

We now prove the scheme is CCA2-secure. Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ and $\widetilde{\Pi} = (\mathsf{Gen}, \widetilde{\mathsf{Enc}}, \widetilde{\mathsf{Dec}})$, where $\widetilde{\mathsf{Enc}}$ and $\widetilde{\mathsf{Dec}}$ are the same as Enc and Dec, except that we replace F_k with a random permutation π. Let E denote the set of random strings used to answer the attack's encryption-oracle queries, and let D denote the set of $n/2$-bit suffixes in the answers to the attacker's decryption-oracle queries. Let $\mathcal{A}$ be an adversary and let $q(\cdot)$ be a polynomial upper-bounding the running-time of $\mathcal{A}$. We claim that:

$$\Pr[\mathsf{PrivK}^{\mathsf{cca2}}_{\mathcal{A},\widetilde{\Pi}}(\lambda) = 1] \leq \frac{1}{2} + \frac{2q(\lambda)}{2^{\lambda/2}}.$$

Let r_c denote the random string used to generate the challenge ciphertext $c = \pi(m \parallel r)$. There are two cases:

1. The value r_c is equal to some element in $E \cup D$. In this case, $\mathcal{A}$ can know which message was encrypted, but the probability of this event occurring is upper-bound by $2q(\lambda)/2^{\lambda/2}$ (this is obtained by applying the union bound).

2. The value r_c is not equal to any of the elements in $E \cup D$. In this case, $\mathcal{A}$ learns nothing about the plaintext because the challenge ciphertext is a uniform string (subject to being distinct from all other ciphertexts).

Let Repeat denote the event that r_c is equal to some element in $E \cup D$. As just discussed, the probability of Repeat is at most $2q(\lambda)/2^{\lambda/2}$, and the probability that $\mathcal{A}$ succeeds in $\mathsf{PrivK}^{\mathsf{cca2}}_{\mathcal{A},\widetilde{\Pi}}$ if Repeat does not occur is exactly $1/2$. Therefore:

$$\begin{aligned}
&\Pr[\mathsf{PrivK}^{\mathsf{cca2}}_{\mathcal{A},\widetilde{\Pi}}(\lambda) = 1] \\
&= \Pr[\mathsf{PrivK}^{\mathsf{cca2}}_{\mathcal{A},\widetilde{\Pi}}(\lambda) = 1 \wedge \mathsf{Repeat}] + \Pr[\mathsf{PrivK}^{\mathsf{cca2}}_{\mathcal{A},\widetilde{\Pi}}(\lambda) = 1 \wedge \overline{\mathsf{Repeat}}] \\
&\leq \Pr[\mathsf{Repeat}] + \Pr[\mathsf{PrivK}^{\mathsf{cca2}}_{\mathcal{A},\widetilde{\Pi}}(\lambda) = 1 \mid \overline{\mathsf{Repeat}}] \\
&\leq \frac{2q(\lambda)}{2^{\lambda/2}} + \frac{1}{2},
\end{aligned}$$

proving our claim. As in our textbook, we can construct a distinguisher $\mathcal{D}$ that determines whether its oracle is pseudorandom or random, by emulating experiment $\mathsf{PrivK}^{\mathsf{cca2}}$ for $\mathcal{A}$ and generating a random bit b for $\mathcal{A}$ to guess. We have $\Pr[\mathcal{D}^{F_k(\cdot)}(1^\lambda) = 1] = \Pr[\mathsf{PrivK}^{\mathsf{cca2}}_{\mathcal{A},\Pi}(\lambda) = 1]$ and $\Pr[\mathcal{D}^{\pi(\cdot)}(1^\lambda) = 1] = \Pr[\mathsf{PrivK}^{\mathsf{cca2}}_{\mathcal{A},\widetilde{\Pi}} = 1]$. Pseudorandomness of F_k implies that

$$\left|\Pr\left[\mathcal{D}^{F_k(\cdot)}(1^\lambda) = 1\right] - \Pr\left[\mathcal{D}^{\pi(\cdot)}(1^\lambda) = 1\right]\right| \leq \mathrm{negl}(\lambda).$$

Combining this with the above claim shows that

$$\Pr\left[\mathsf{PrivK}^{\mathsf{cca2}}_{\mathcal{A},\Pi}(\lambda) = 1\right] \leq \frac{1}{2} + \frac{2q(\lambda)}{2^{\lambda/2}} + \mathrm{negl}(\lambda),$$

thus we conclude that Π is CCA2-secure.

Problem 5.

Part A.

$$H_1(k_1, H_2(k_2, m_0)) = H_1(k_1, H_2(k_2, m_1))$$

Then either $H_2(k_2, m_0) = H_2(k_2, m_1)$, or $H_1(k_1, m_0') = H_1(k_1, m_1')$, here $m_0' = H_2(k_2, m_0)$, $m_1' = H_2(k_2, m_1)$, $m_0' \neq m_1'$.

$$\begin{aligned}
&\Pr[H_1(k_1, H_2(k_2, m_0)) = H_1(k_1, H_2(k_2, m_1))] \\
&\leq \Pr[H_2(k_2, m_0) = H_2(k_2, m_1)] + \Pr[H_1(k_1, m_0') = H_1(k_1, m_1')] \\
&\leq \varepsilon_1 + \varepsilon_2
\end{aligned}$$

Part B. If $\delta \neq 0$, let $m_0' = H_2(k_2, m_0)$, $m_1' = H_2(k_2, m_1)$. $H_1(k_1, m_0') - H_1(k_1, m_1') = \delta$ implies $m_0' \neq m_1'$,

$$\begin{aligned}
&\Pr[H_1(k_1, H_2(k_2, m_0)) - H_1(k_1, H_2(k_2, m_1)) = \delta] \\
&= \Pr[H_1(k_1, m_0') - H_1(k_1, m_1') = \delta] \leq \varepsilon_2
\end{aligned}$$

If $\delta = 0$ then either $H_2(k_2, m_0) = H_2(k_2, m_1)$, or $H_1(k_1, m_0') = H_1(k_1, m_1')$, here $m_0' = H_2(k_2, m_0)$, $m_1' = H_2(k_2, m_1)$, $m_0' \neq m_1'$. Using the result in part A,

$$\Pr[H_1(k_1, H_2(k_2, m_0)) - H_1(k_1, H_2(k_2, m_1)) = 0] \leq \varepsilon_1 + \varepsilon_2$$