1. In basic CBC-MAC, \( t_0 \) is fixed. Show that the following modification (where \( t_0 \) is not fixed) does not yield a secure fixed-length MAC for messages of length \( nq \).

(Modified) Tag generation: For key \( k \in \{0,1\}^n \) and message \( m \in \{0,1\}^{nq} \),

- parse \( m \) as \( m = (m_1, \ldots, m_q) \) // \( q \) blocks //
- apply CBC to \( m \), i.e., let

\[
t_0 \leftarrow \{0,1\}^n \quad \text{and} \quad t_i := F_k(m_i \oplus t_{i-1}) \quad \text{for} \quad 1 \leq i \leq q
\]

- output \( \langle t_0, t_q \rangle \) as the tag

2. Show that appending the message length \( |m| \) to the end of the message \( m \) before applying basic CBC-MAC does not result in a secure MAC for arbitrary-length messages.

(Assume \( |m| < 2^n \).)

3. Let \( F \) be a pseudorandom function. Construct a fixed-length MAC scheme for messages of length \( 2n \) as follows. The shared key is a random \( k \in \{0,1\}^n \). To authenticate a message \( m_1, m_2 \) with \( |m_1| = |m_2| = n \), let the tag be \( \langle F_k(m_1), F_k(F_k(m_2)) \rangle \). Is this scheme secure against chosen-message attacks? Justify your answer.