Problem 4): Begin by defining $\prod$ to be the encryption scheme $\prod = (\ \texttt{Gen}, \texttt{Enc}, \texttt{Dec}\ )$ where

- The security parameter $n \in \mathbb{Z}$ and $\texttt{Gen}$ are used to generate the key k by running $\texttt{Gen}\left(1^n\right) = k$
- The key k, message m, and $\texttt{Enc}$ are used to produce cipher-text c by running $\texttt{Enc}_k\left(m\right) = c$
- The key k, cipher-text c, and $\texttt{Dec}$ are used to recover the message m by running $\texttt{Dec}_k\left(c\right) = m$

Additionally, let m_0, m_1 be messages of the same length and c be the cipher-text generated from one of the messages by running $\texttt{Enc}_k\left(m_b\right) = c$, where $b = \{0, 1\}$. The encryption scheme $\prod$ is considered to be **CPA** secure if the probability of a polynomial time-limited adversary $\mathcal{A}$, with access to m_0, m_1 and c, determining which message was used to compute c is equal to the sum of 1/2 and any value that is negligible on the order of n.

Now denote the experiment above as $\texttt{Priv}^{\text{CPA}}_{\mathcal{A},\prod}\left(n\right)$. Let this return 0 except when $\mathcal{A}$ is able to determine which message was used to compute c then let $\texttt{Priv}^{\text{CPA}}_{\mathcal{A},\prod}\left(n\right)$ return 1. Using this notation, our definition **CPA** security can be formally stated

$$\Pr\left[\texttt{Priv}^{\text{CPA}}_{\mathcal{A},\prod}\left(n\right) = 1\right] \leq \frac{1}{2} + \ \texttt{negl}\ \left(n\right) \tag{4.1}$$

where $\texttt{negl}\left(n\right)$ is a negligible function of order n.

Finally, consider the case for the experiment $\texttt{Priv}^{\text{CPA}}_{\mathcal{A},\prod}\left(n\right)$ where the messages m_0, m_1 passed to the adversary $\mathcal{A}$ are such that $m_0 = m_1$ and the result of $\texttt{Enc}_k\left(m_i\right) = c_i$ is fixed each m_i in the message space. That is to say, for any fixed k, that each $c_i \in \mathcal{C}$ is determined by the result of $\texttt{Enc}_k\left(m_i\right)$ for only one $m_i \in \mathcal{M}$. In this case the result of $\texttt{Priv}^{\text{CPA}}_{\mathcal{A},\prod}\left(n\right)$ will always be 1 because the cipher-texts $c_0 = \texttt{Enc}_k\left(m_0\right)$ and $c_1 = \texttt{Enc}_k\left(m_1\right)$ are always equal thereby allowing $\mathcal{A}$ *always* to succeed every time this is case. In this case, $\prod$ does not satisfy the definition given in expression 4.1 and is therefore not **CPA** secure. Thus, we must impose an additional requirement on **CPA** secure encryption schemes.

The problem arises from the case when $m_0 = m_1$ and the when, for each fixed $m_i \in \mathcal{M}$, the function $\texttt{Enc}_k\left(m_i\right)$ always returns the the same c_i. That is to say that the operation $\texttt{Enc}_k\left(m_i\right)$ on each $m_i \in \mathcal{M}$ always determines single, unique corresponding $c_i \in \mathcal{C}$. With this in mind, we refine our definition of

CPA security to also include the requirement that, given a fixed key k, the `Dec` algorithm be non-deterministic on $m \in \mathcal{M}$. This is equivalent to requiring that the `Dec` algorithm be such that any passed $m_i \in \mathcal{M}$ can return any $c \in \mathcal{C}$ with some non-zero probability, thereby making `Dec` probabilistic instead of deterministic.