Problem 1): We know that the expression

$$\Pr\left[M = m \,|\, C = c\right] = \Pr\left[M = m\right], \tag{1.1}$$

which holds for some encryption scheme (`Gen`, `Enc`, `Dec`). We apply Bayes' formula to the left-hand-side of expression 1.1 to yield

$$\Pr\left[M = m \,|\, C = c\right] = \frac{\Pr\left[M = m, C = c\right]}{\Pr\left[C = c\right]} \tag{1.2}$$

Since the $\Pr\left[M = m, C = c\right]$ term in expression 1.2 represents a Joint Probability Distribution, we also have

$$\Pr\left[M = m, C = c\right] = \Pr\left[M = m\right]\,\Pr\left[C = c \,|\, M = m\right]$$

This allows the result in expression 1.2 to be equivalently expressed as

$$\Pr\left[M = m \,|\, C = c\right] = \frac{\Pr\left[M = m\right]\,\Pr\left[C = c \,|\, M = m\right]}{\Pr\left[C = c\right]}$$

which is divided by $\Pr\left[M = m\right]$ to give

$$\frac{\Pr\left[M = m \,|\, C = c\right]}{\Pr\left[M = m\right]} = \frac{\Pr\left[C = c \,|\, M = m\right]}{\Pr\left[C = c\right]} \tag{1.3}$$

By noting that expression 1.1 implies

$$\frac{\Pr\left[M = m \,|\, C = c\right]}{\Pr\left[M = m\right]} = 1,$$

the result in expression 2.2 becomes

$$\frac{\Pr[C = c \,|\, M = m]}{\Pr[C = c]} = 1$$

Multiplying this result by $\Pr[C = c]$ gives

$$\Pr[C = c \,|\, M = m] = \Pr[C = c], \tag{1.4}$$

thereby proving that $\Pr[M = m \,|\, C = c] = \Pr[M = m]$ implies $\Pr[C = c \,|\, M = m] = \Pr[C = c]$.
$\square$

Problem 2): For a message of length $l \in \mathbb{Z}^+$, a one-time pad will be associated with three separate set spaces and three different algorithms.

The set spaces of a one-time pad are the key space $\mathcal{K}$, the message space $\mathcal{M}$, and the cipher-text space $\mathcal{C}$. Additionally these spaces are such that

$$\mathcal{K} = \mathcal{M} = \mathcal{C} = \{0,1\}^l, \tag{2.1}$$

where $\{0,1\}^l$ is the set of all binary strings having length l. Formally, the set $\{0,1\}^l$ is defined $\{0,1\}^l \equiv \{n_1, n_2, \ldots, n_l\}$ where $\forall i \in [1,l], n_i \ni [0,1] \subset \mathbb{Z}$ (i.e. either $n_i = 0$, or $n_i = 1$ for every element in the set)[1].

A one-time pad is also associated with the algorithms

- The key-generation algorithm, denoted `Gen`
- The encryption algorithm, denoted `Enc`

[1]This also holds for $\mathcal{M}$ and $\mathcal{C}$

- The decryption algorithm, denoted `Dec`

The purpose of `Gen` is to generate a key for encrypting and decrypting our message, where ee denote this key k and say that $k \in \mathcal{K}$. We say that `Gen` works by choosing a string from $\mathcal{K}$ according to the uniform distribution. From this choice of distribution, it follows that each possible key will be chosen with probability 2^{-l}.

Before we describe `Enc` and `Dec` algorithms we must define a bit-wise **XOR** on two binary strings of equal length. Let a and b be any two binary strings such that $a, b \in \{0,1\}^l$. Additionally, let us express these strings by

$$a \equiv \{a_1, a_2, \ldots, a_l\}$$

and

$$b \equiv \{b_1, b_2, \ldots, b_l\},$$

respectively. The bit-wise **XOR** of a and b is be denoted by $a \oplus b$ and expressed as the binary string

$$a \oplus b \equiv \{a_1 \oplus b_1, a_2 \oplus b_2, \ldots, a_l \oplus b_l\}$$

The elements of this binary string, the $a_i \oplus b_i$, are binary bits and are defined $\forall i \in [1, l]$ to be the traditional bit-level **XOR** of a_i and b_i, denoted $a_i \oplus b_i$. We use the truth table

a_i	b_i	$a_i \oplus b_i$
0	0	0
0	1	1
1	0	1
1	1	0

to express the definition of the transitional bit-level **XOR** and continue to the definitions of the `Enc` and `Dec` algorithms.

The `Enc` algorithm is used to encrypt the message into cipher-text based on the chosen key. We will denote the message to be encrypted by m, the cipher-text resulting from the encryption by c, and the key by k. These will each be such that $m \in \mathcal{M}$, $c \in \mathcal{C}$, and $k \in \mathcal{K}$. Using this notation and our definition for bitwise **XOR** from above, we define

$$c := m \oplus k$$

to be the expression used by `Enc` as it encrypts the message.

Inversely from the `Enc` algorithm, the `Dec` algorithm is used to decrypt the cipher-text back into the message based on the supplied key. Similarly to `Enc`, we define

$$m := c \oplus k$$

to be the expression used by `Dec` as it decrypts the message.

To prove the security of this one-time pad, consider any arbitrary message m and any arbitrary cipher-text c, where $m \in \mathcal{M}$ and $c \in \mathbb{C}$. Now, we express the probability of finding a particular c, given a particular m by

$$\Pr\left[C = c \,|\, M = m\right] \tag{2.2}$$

Using the fact that $c = m \oplus k$, this expression may be rewritten as

$$
\begin{aligned}
\Pr\left[C = c \,|\, M = m\right] &= \Pr\left[M \oplus K = c \,|\, M = m\right] \\
&= \Pr\left[M \oplus K = c \,|\, M = m\right] = \Pr\left[m \oplus K = c\right]
\end{aligned}
$$

Next, we **XOR** the random variable term in this expression by m to obtain

$$
\Pr\left[m \oplus (m \oplus K) = m \oplus c\right] = \Pr\left[K = m \oplus c\right],
$$

because $a \oplus a = 0$ for any binary string a. Above, we defined the probability of choosing any k to be $\Pr\left[K = k\right] = 2^{-l}$. This allows us to finally obtain the relations

$$
\begin{aligned}
\Pr\left[C = c \,|\, M = m\right] &= \Pr\left[M \oplus K = c \,|\, M = m\right] \\
&= \Pr\left[m \oplus K = c\right] \\
&= \Pr\left[m \oplus (m \oplus K) = m \oplus c\right] \\
&= \Pr\left[K = m \oplus c\right] = \frac{1}{2^l}
\end{aligned} \tag{2.3}
$$

Since or choice of m in expression 2.2 was arbitrary, the result in expression 2.3 must hold for any $m \in \mathcal{M}$. This implies that, for any $m_0, m_1 \in \mathcal{M}$, we relation

$$
\Pr\left[K = m_0 \oplus c\right] = \frac{1}{2^l} = \Pr\left[K = m_1 \oplus c\right]
$$

holds. This satisfies *Lemma 2.3* from the text, thus the one-time pad is perfectly secure.