2) let us assume $K = 2$ and $M_1 \neq M_2$ where M stands for the message.

$$C_1 = K \oplus M_1$$
$$C_2 = K \oplus M_2$$

$\rightarrow C_1$ stands for the ciphertext
- same key is used to generate ciphertexts

→ for the program in part (1), we have C_1, C_2, M_1 and M_2.

$$M_1 \oplus C_1 = X_1$$
$$M_1 \oplus C_2 = X_2$$
$$M_2 \oplus C_1 = X_3$$
$$M_2 \oplus C_2 = X_4$$

(*) If we use the same key to encrypt M_1 and M_2, two of $\{X_1, X_2, X_3, X_4\}$ has to be the same

Because;

$$M_1 \oplus C_1 \equiv M_1 \oplus (M_1 \oplus K)$$

$$\equiv K$$

Note (1): $M_1 \oplus M_1 = 0$ //
$0 \oplus x = x$ //

Note (2): $a \oplus (b \oplus c)$
$\Downarrow\Downarrow$
$(a \oplus b) \oplus c$

(since XOR holds associative property

In my program, I tried to find common key by comparing x_1, x_2, x_3, x_4. If two of x_1, x_2, x_3, x_4 are equal, the equal ones will be the key. The probability to find true key where all plaintexts are encrypted with same key is $\frac{1}{K}$ for $K > 1$.

- If plaintexts are the same and all ciphertexts are generated with same key,

$$x_1 = x_2 = x_3 = x_4.$$