Questions 1-2 are similar to yesterday. Make sure you have a written solution to at least one part of each problem, and ask Conrad or Aaron to check it. Try to make sure you understand how perfect secrecy works, but don't spend more than an hour on this.

You can use the following way to check perfect secrecy: A cipher has perfect secrecy if every messages is encrypted to the same ciphertext with the same probability. That is:

Definition 1 *(A cipher with the encryption function E has perfect secrecy if) for every $m \in \mathcal{M}$ and $c \in \mathcal{C}$, the probability*

$$Pr[E(k \xleftarrow{R} \mathcal{K}, m) = c]$$

is always the same (or doesn't depend on m).

Make sure to prove that something has perfect secrecy by doing this calculation, or to disprove it by showing two messages $m_0 \in \mathcal{M}$ such that this doesn't hold.

1 One-Time Pad Overkill

Suppose we use a double-key for OTP:

$$\mathcal{K} = \mathcal{K}_1 \times \mathcal{K}_2 = \{0,1\}^n \times \{0,1\}^n$$
$$\mathcal{M} = \{0,1\}^n$$
$$\mathcal{C} = \{0,1\}^n \times \{0,1\}^n$$

- $E((k_1, k_2), m) = (k_1 \oplus m, k_2 \oplus m)$
- $D((k_1, k_2), (c_1, c_2)) = k_1 \oplus c_1$

Prove that this cipher has perfect secrecy. (Make sure to use the definition of perfect secrecy.)

[SOLUTION: For any $m \in \mathcal{M}, c \in \mathcal{C}$:

$$Pr[E(k \xleftarrow{R} \mathcal{K}, m) = c] = \frac{1}{(2^n)^2}$$

Thus, the definition of perfect secrecy holds if we compare any two messages.]

2 Analyze that Cipher!$^{\text{TM}}$

Suppose we have a cipher

- $E : \mathcal{K} \times \mathcal{M} \to \mathcal{C}$
- $D : \mathcal{K} \times \mathcal{C} \to \mathcal{M}$

that has perfect secrecy. Consider the following ciphers that are built on it. For each one:

- Verify that correctness holds (show that encrypting and decrypting a message m using the same key always gives you back m).
- Figure out if it has perfect secrecy. Prove/disprove it.

2.1

- $\mathcal{K}' = \mathcal{K}$
- $\mathcal{M}' = \mathcal{M}$
- $\mathcal{C}' = \mathcal{C} \times \{0, 1\}$
- $E'(k, m) = (E(k, m), m[0])$
- $D'(k, (c_1, c_2) = D(k, c_1)$

2.2

- $\mathcal{K}' = \mathcal{K}$
- $\mathcal{M}' = \mathcal{M}$
- $\mathcal{C}' = \mathcal{C} \times \mathcal{K}$
- $E'(k, m) = (E(k, m), k)$
- $D'(k, (c_1, c_2) = D(k, c_1)$

[SOLUTION: This does not have perfect secrecy.

Consider an encryption $c = (c_1, c_2) = E(k \xleftarrow{R} \mathcal{K}, m)$ of a message m. This means that $k = c_2$.

Suppose another message m' can encrypt to the same ciphertext. Since

$$E(k', m') = c = (c_1, c_2)$$

this means that $k' = c_2 = k$.

However, we also know that $D(k, c) = m$. Thus, two messages cannot encrypt to the same ciphertext, and we definitely don't have perfect secrecy.

]

3

3.1

Consider a "modular" version of the one-time pad.

- $\mathcal{K} = \mathbb{Z}_n^*$
- $\mathcal{M} = \mathbb{Z}_n^*$
- $\mathcal{C} = \mathbb{Z}_n^*$
- $E(k, m) = m + k$
- $D(k, c) = c - k$

Prove that it has perfect secrecy.

3.2

Do the same thing for the multiplicative group $(modn)$ Consider a "modular" version of the one-time pad.

- $\mathcal{K} = \mathbb{Z}_n^*$
- $\mathcal{M} = \mathbb{Z}_n^*$
- $\mathcal{C} = \mathbb{Z}_n^*$

- $E(k, m) = m \cdot k$
- $D(k, c) = c \cdot k^{-1}$

Prove that it has perfect secrecy.

4 Functions

Recall:

$$\Big|FUNCTIONS[X \to Y]\Big| = |Y|^{|X|}$$

In particular, keep in mind that the function is determined by $|X|$ choices of values from Y, so $|X|$ is in the exponent.

4.1

Consider the functions $f : \mathbb{Z}_n \to \mathbb{Z}_n$ for $n = 2$. How many such functions are there?

$$\Big|FUNCTIONS[\mathbb{Z}_n \to \mathbb{Z}_n]\Big|$$

Write down a description of each function $\in FUNCTIONS[\mathbb{Z}_n \to \mathbb{Z}_n]$

4.2

Calculate how meany of each of the following kinds of functions there are:

X	Y	$\Big\lvert FUNCTIONS[X \to Y]\Big\rvert$
$\mathbb{Z}_p$	$\mathbb{Z}_p$	
$\mathbb{Z}_n$	$\mathbb{Z}_n^*$	
{meow, purr}	{happy, sad, hungry}	
{happy, sad, hungry}	{meow, purr}	

5 One-Time Pad Malleability

Alice is sending her bank the message “SEND $1000 TO BOB”. Here is the encryption using a stream cipher:

```
00011011 00101110 11011011 10111011 00000100
10000111 11010101 10001111 10010000 01101111
00100011 11111010 11111001 10110100 11011101
01010000 11100000
```

Modify the message so that it decrypts to “SEND $1000 TO EVE” instead (using the same key). You don’t have to rewrite the whole message, just the part that changes.

Here’s a table of ASCII values for the uppercase letters again:

A	65	01000001		N	78	01001110
B	66	01000010		O	79	01001111
C	67	01000011		P	80	01010000
D	68	01000100		Q	81	01010001
E	69	01000101		R	82	01010010
F	70	01000110		S	83	01010011
G	71	01000111		T	84	01010100
H	72	01001000		U	85	01010101
I	73	01001001		V	86	01010110
J	74	01001010		W	87	01010111
K	75	01001011		X	88	01011000
L	76	01001100		Y	89	01011001
M	77	01001101		Z	90	01011010

[SOLUTION: Key: The last three bytes change to

$$\texttt{11111001 10110100 11011101} \oplus \text{“}\texttt{EVE}\text{''} \oplus \text{“}\texttt{BOB}\text{''}$$

$$= \texttt{11011010 01001001 11100111}$$

]