1. The sequence of bits $s_0, s_1, s_2, \ldots$ is the output of this nonlinear feedback shift register:

   State: A 3-bit register $s = s_2 s_1 s_0$.
   Init: Set s to equal the 3-bit key $k = k_2 k_1 k_0$.
   Next: Set *next* to $s_2 + s_1 s_0 \bmod 2$.
   Output s_0.
   Right shift all other bits of s.
   Set the leftmost bit of s to *next*.

   (a) Calculate the first 10 bits of the output y when the cipher is instantiated with $k_2 k_1 k_0 = \texttt{010}$ and $\texttt{100}$.

y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9
0	1	0							
0	0	1							

   Solution:

y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9
0	1	0	0	0	0	0	0	0	0
0	0	1	1	1	0	1	1	1	0

   (b) Alice encrypts m as $c = m + y$. Eve observes ciphertext $c = \texttt{1010100111}$. Circle the only possible m among the following alternatives. Explain why m is possible but the other two are not.

   `0000000000` `1001001001` `1111111111`

   Solution: $\boxed{\texttt{1001001001}}$. As all possible states show up in part (a), the encryptions of the first and third message must repeat in blocks of four (if the first two bits are discounted). For instance, c_5 would have to equal c_9. This is inconsistent with the first and third message. On the other hand, c is a possible encryption of the second message with key $k_2 k_1 k_0 = \texttt{100}$.

   Alternative solution: If c was a possible encryption of all zeros, the key would have to be the first 3 bits of c, namely `101`, the next bit y_4 would be a `1`, which is inconsistent with $y = c$. Similarly, if c was an encryption of all ones, the key would have to be `010`, the next bit y_4 a `0`, which is again inconsistent with $y = c + \texttt{1111}\cdots$.

2. The secret key to a pseudorandom permutation P on 5-bit strings is known to Alice and Bob. To encrypt a 5-bit message m, Alice chooses a random 5-bit pad x and sends $(P(x), x + m)$.

   (a) Bob receives ciphertext (y, c). How should he decrypt it?

   Solution: Bob first applies P^{-1} to y and then unpads c: $m = P^{-1}(y) + c$.

   (b) Alice encrypts $m_1 = \texttt{00000}$. She then encrypts $m_2 = \texttt{00000}$ again. What is the probability that the resulting ciphertexts are identical?

   Solution: The two encryptions have the form $(P(x_1), x_1)$ and $(P(x_2), x_2)$. The ciphertexts are equal exactly when x_1 equals x_2. This happens with probability $1/2^5 = 1/32$.

   (c) How does your answer to part (b) change when $m_1 = \texttt{00000}$ and $m_2 = \texttt{11111}$?

   Solution: The probability is zero.The two encryptions have the form $(P(x_1), x_1)$ and $(P(x_2), x_2 + \texttt{11111})$. These ciphertexts are never equal: If $x_1 = x_2 + \texttt{11111}$, then $x_1 \neq x_2$, so $P(x_1) \neq P(x_2)$.

3. The secret key to a pseudorandom function F with 3-bit input and output is known to Alice and Bob. To encrypt a 6-bit message $m = m_L m_R$, Alice chooses a random 3-bit pointer x and sends $(x,\ m_L + F(x),\ m_R + F(m_L))$.

   (a) Bob receives ciphertext $(\texttt{001}, \texttt{011}, \texttt{100})$. What does it decrypt to? F is given by

x	000	001	010	011	100	101	110	111
$F(x)$	011	000	100	101	011	100	110	000

   Solution: $F(x) = \texttt{000}$, so $m_L = \texttt{011}$, $F(m_L) = \texttt{101}$, and $m_R = \texttt{001}$. It decrypts to 011001.

   (b) Can Eve distinguish an encryption of 000000 from 9 pure random bits? Explain. **yes no**

   Solution: Yes. Here is one distinguisher: Eve accepts if $x = \texttt{000}$ and the middle three bits c_L and last three bits c_R are equal. For an encryption of 000000, $x = \texttt{000}$ implies $c_L = c_R$, so Eve accepts with probability 1/8. For 9 truly random bits, the events $x = \texttt{000}$ and $c_L = c_R$ are independent, and each has probability 1/8. The overall probability is $1/8^2$. The distinguishing advantage is $1/8 - 1/8^2 = 7/64$.

   (c) Alice encrypts one message followed by another. Find m_1, m_2 so that Eve can distinguish encryptions of

$$m_1 = _____,\quad m_2 = _____ \qquad \text{from encryptions of} \qquad m_1' = \texttt{000000},\ \ m_2' = \texttt{111111}$$

   with advantage at least 7/8. Explain how the distinguisher works.

   Solution: For example, choose $\boxed{m_1 = m_2 = \texttt{000000}}$. For these two messages, the right ciphertext blocks in both encryptions are $F(\texttt{000})$, so they are always equal. Eve therefore distinguishes by testing equality of the two right ciphertext blocks.

   For the comparison messages $m_1' = \texttt{000000}$ and $m_2' = \texttt{111111}$, the right blocks are $F(\texttt{000})$ and $\texttt{111} + F(\texttt{111})$. If F is replaced by a truly random function, these are equal with probability 1/8. The equality test therefore has distinguishing advantage $1 - 1/8 = 7/8$.