CSE 130 Homework 1 Solutions (Spring 2025)
Jeffrey Peng

Historical Ciphers:

1. One possible way is to use the Index of Coincidence. The primary difference between a shift cipher and a Vigenere cipher is that a shift cipher is more simple in encrypting (26 possible shifts) compared to a Vigenere cipher's dependency on key lengths, frequency analysis, etc. As shown in our lab, a Vigenere cipher's key could also be in bits as well. Due to this, the Index of Coincidence will likely be close to the standard characteristic of English, or 0.067. But in a Vigenere Cipher, due to the exponentially larger possibilities, the index of coincidence will likely be lower.
2. The index of coincidence method would not work relying on $\sum_i p_i$ despite the actual equation being $\sum_i p_i^2$. This is because of the index of coincidence's reliance on probability, specifically, the probability of two randomly selected letters being the same. If we were to use $\sum_i p_i$, the sum of all probabilities is always going to be equal to 1. However, since we are trying to measure the variance and/or concentration of the distribution, the square is needed to measure how spread out or concentrated the probabilities are.
3. Since $\sum_i p_i^2 = 0.067$ For English, if we knew that the plaintext wasn't English, we would likely have to adjust the index of coincidence's values to be whatever values for the language. We would need to adjust both the letter frequency distribution (p_i) as well as be ready to receive a different index of coincidence value that isn't the English value (0.067).

Modern Cryptography, Perfect Secrecy and One Time Pad:

1. We have $Pr[M = m|C = c] = Pr[M = m'|C = c]$.
   - This is attempting to be "For every perfectly secret encryption scheme it holds that for every distribution on the message space M, every m m' ∈ M, and every c ∈ C
   - However, for perfect secrecy, the ciphertext should not change the attacker's knowledge about the distribution of M, meaning,
     - $Pr[M = m|C = c] = Pr[M = m]$
   - If we observe the left, $Pr[M = m|C = c] = Pr[M = m]$
   - Observing the right, $Pr[M = m'|C = c] = Pr[M = m']$
     - We therefore have Pr[m] = Pr[m'] which would mean that the probability of message m appearing is the same as message m' appearing. The problem with this is that the English language is not uniform so every message does not have the same probability of appearing. This therefore makes $Pr[M = m|C = c] = Pr[M = m'|C = c]$ **False.**
2. We first need to convert the plaintext and key from hexadecimal to binary.

Plaintext	0x012345

0x01	0000 0001
0x23	0010 0011
0x45	0100 0101

Plaintext	0xFFEEDD
0xFF	1111 1111
0xEE	1110 1110
0xDD	1101 1101

We now need to Perform Bitwise XOR

Plaintext	Key	XOR Result
0000 0001	1111 1111	1111 1110
0010 0011	1110 1110	1100 1101
0100 0101	1101 1101	1001 1000

Converting those back to Hex

1111 1110	0xFE
1100 1101	0xCD
1001 1000	0x98

Therefore, our final answer is 0xFECD98

3. To determine perfect secrecy, we need the probability distribution of the ciphertext to be independent of the plaintext. We are given: $Enc_k(m) \ = [m + k \, mod \, 3]$

   The following answers work is shown below:

   A. Not Secret

   B. Perfectly Secret

   C. Perfectly Secret

   - The message space is M = {0,1}, and Gen chooses a uniform key from the key space K = {0,1}
     - If m = 0:

       $k \ = \ Enc_0(0) \ = \ [0 \ + \ 0 \, mod \, 3] \ = \ 0$

       $k \ = \ Enc_1(0) \ = \ [0 \ + \ 1 \, mod \, 3] \ = \ 1$

- ➢ If m = 1:

  $k = Enc_0(1) = [1 + 0\,mod\,3] = 1$

  $k = Enc_1(1) = [1 + 1\,mod\,3] = 2$

- ➢ The possible ciphertexts for this message space are {0,1,2}. From looking at the distribution, we can see that we can **only** get 2 at m=1 and k=1 instead of getting 2 equally for any plaintext. Therefore, this message space and uniform key **is not perfectly secret.**

- ○ The message space is M = {0, 1, 2}, and Gen chooses a uniform key from the key space K = {0,1,2}
  - ➢ If m=0:

    $k = Enc_0(0) = [0 + 0\,mod\,3] = 0$

    $k = Enc_1(0) = [0 + 1\,mod\,3] = 1$

    $k = Enc_2(0) = [0 + 2\,mod\,3] = 2$
  - ➢ If m=1:

    $k = Enc_0(1) = [1 + 0\,mod\,3] = 1$

    $k = Enc_1(1) = [1 + 1\,mod\,3] = 2$

    $k = Enc_2(1) = [1 + 2\,mod\,3] = 0$
  - ➢ If m=2:

    $k = Enc_0(2) = [2 + 0\,mod\,3] = 2$

    $k = Enc_1(2) = [2 + 1\,mod\,3] = 0$

    $k = Enc_2(2) = [2 + 2\,mod\,3] = 1$
  - ➢ The possible ciphertexts for this message space are {0,1,2}. From looking at the distribution, we can see that we equally get 0, 1, or 2 for all message space M and uniform key K. Therefore, this message space and uniform key are **Perfectly Secret.**

- ○ The message space is M = {0,1}, and Gen chooses a uniform key from the key space K = {0,1,2}
  - ➢ If m=0:

    $k = Enc_0(0) = [0 + 0\,mod\,3] = 0$

    $k = Enc_1(0) = [0 + 1\,mod\,3] = 1$

    $k = Enc_2(0) = [0 + 2\,mod\,3] = 2$
  - ➢ If m=1:

    $k = Enc_0(1) = [1 + 0\,mod\,3] = 1$

    $k = Enc_1(1) = [1 + 1\,mod\,3] = 2$

    $k = Enc_2(1) = [1 + 2\,mod\,3] = 0$
  - ➢ The possible ciphertexts for this message space are {0,1,2}. From looking at the distribution, we can see that we equally get 0, 1, or 2 for all

message space M and uniform key K. Therefore, this message space and uniform key are **Perfectly Secret.**

4. $K\ =\ \{0, 1\}^l\ and\ Enc_k(m)\ outputs\ k_{|m|}\oplus\ m$ where k_t denotes the first *t* bits of *k*.
   - Assume 2 distinct messages $m_1\ and\ m_2$ of the same length *t*. These messages will be encrypted by $Enc_k(m_1)\ =\ k_t\oplus m_1$ and $Enc_k(m_2)\ =\ k_t\oplus m_2$. When looking at these encryptions, the ciphertext is formed by $c\ =\ k_t\oplus m_1$ which also means that the ciphertext can be decoded by $c\oplus m_1\ =\ k_t$. If done by someone attempting to break the cipher, they would now have k_t or the first *t* bits of the key. With this, they would now be able to decode m_2 using the same decode method now that they also have the key. Since the encryption and the ciphertext rely on both the key and message, but then having the ciphertext would make it theoretically possible to obtain the key, making this encryption scheme not perfectly secret for message space M.
   - To make this a perfectly secret encryption scheme for message space M, we can try using the one-time pad approach. Since we have $K\ =\ \{0, 1\}^l$ and $M\ =\ \{0, 1\}^{<=l}$. With the existing scheme, $Enc_k(m)\ =\ k_{|m|}\oplus m$. With the one-time pad approach, we include the key *k* being as long as the message *m* and the key *k* is used only once. Since we know that the key *k is as long* as the maximum message length *l* for a message *m* of length *t* <= l, only the first *t* bits of *k* are used. Additionally, due to OTP, the key *k* is never reused, therefore each message has a new encrypted message and new key. Since the key is fresh every time, the issue we had in part A (where decoding a message would allow you to decode all messages encrypted this way) would no longer happen as each message would have its own independent key.
5. Given: key (*t* = 2), $m_0\ =\ aaa\ and\ m_1\ =\ aab$. Ciphertext *c* outputs 0 if the first character of c is the same as the third character, and 1 if anything else. Compute $Pr[PrivK^{eav}_{A,II}\ =\ 1]$, which represents the probability that ‘A’ will be able to correctly guess if ciphertext c corresponds to $m_0\ or\ m_1$.
   - Since we know that the key is *t*=2, we know that the key repeats every 2 key characters.
     - We therefore have:
     - $c_1 = m_1 + k_1\%\ 26$
     - $c_2 = m_2 + k_2\%\ 26$
     - $c_3 = m_3 + k_1\%\ 26$
   - For the messages we have $m_0\ and\ m_1$
     - $m_0 =\ "aaa"$
       - $c_1\ =\ a\ +\ k_1$

- $c_2 = a + k_2$
- $c_3 = a + k_1$
    - $So\ c_1 = c_3$
- $m_1 = "aab"$
    - $c_1 = a + k_1$
    - $c_2 = a + k_2$
    - $c_3 = b + k_1$
        - $So\ c_1 ! = c_3$
- Knowing this, if ‘A’ receives a ciphertext c, They see that $c_1 = c_3$ so they put output as 0. Otherwise, they put 1.
- Since the key is random and therefore uniform for $k_1\ and\ k_2$ over the 26 letters, then we have
    - b=0 meaning m_0 was encrypted which will always have $c_1 = c_3$
        - Therefore ‘A’ outputs 0 with probability 1
    - b=1 meaning m_1 was encrypted which will always have $c_1 ! = c_3$
        - Therefore ‘A’ outputs 1 with probability 1
- **We therefore get** $Pr[PrivK^{eav}_{A,II} = 1] = 1$ **which means that the scheme is completely insecure under the chosen ciphertext attacks.**

6. A).
    - Given 1011 0111 and 1110 0111, we can assume that these ciphertexts were encrypted using a one-time pad (OTP) with the same 8-bit key. With these ciphertexts using OTP, we can apply the XOR of two OTP-encryptions ciphertexts which would cancel out the key:
        - $C_1 \oplus C_2 = (P_1 \oplus K) \oplus (P_2 \oplus K) = P_1 \oplus P_2$
        - $C_1 \oplus C_2$
            - $1011\ 0111 \oplus 1110\ 0111 = 0101\ 0000$
        - Therefore: $P_1 \oplus P_2 = 0101\ 0000$ which corresponds to the character ‘P’ which also means that $P_1\ and\ P_2$ are separated by the ASCII difference of ‘P’ and there’s also space in the ciphertext.
6. B).
    - Given 0110 0110, 0011 0010, 0010 0011
        1. We can use the same thing applied from A to compute the XOR relationships
            - However, we can assume that one of these ciphertext are spaces.
            - Assuming 0110 0110 is the space,
                1. $C_1 \oplus C_2$: $0110\ 0110 \oplus 0011\ 0010 = 0101\ 0100$ = T

b. $C_1 \oplus C_3$: $0110\ 0110 \oplus 0010\ 0011\ =\ 0100\ 0101$ $=$ E

- ➢ Assuming 0011 0010 is the space

   a. $C_2 \oplus C_3$: $0011\ 0010 \oplus 0010\ 0011\ \ =\ \ 0001\ 0001$ = Non-printable

- ➢ Assuming 0010 0011 is the space

   a. $C_3 \oplus C_2$: $0010\ 0011 \oplus 0011\ 0010\ \ =\ \ 0001\ 0001$ = Non-printable

ii. Therefore, we know that $P_1, P_2,\ and\ P_3$ are separated by something with "'space', 'T', 'E'" or " TE".