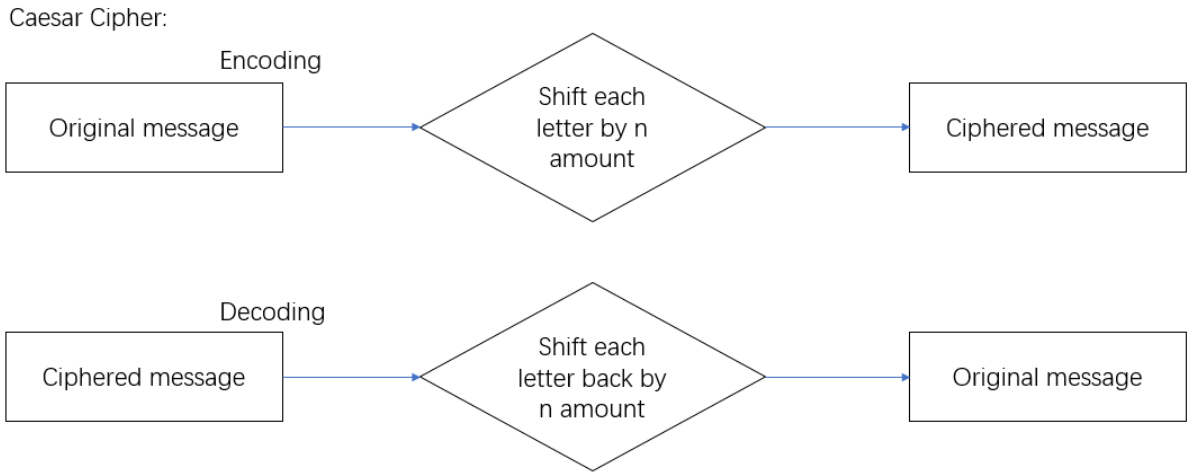
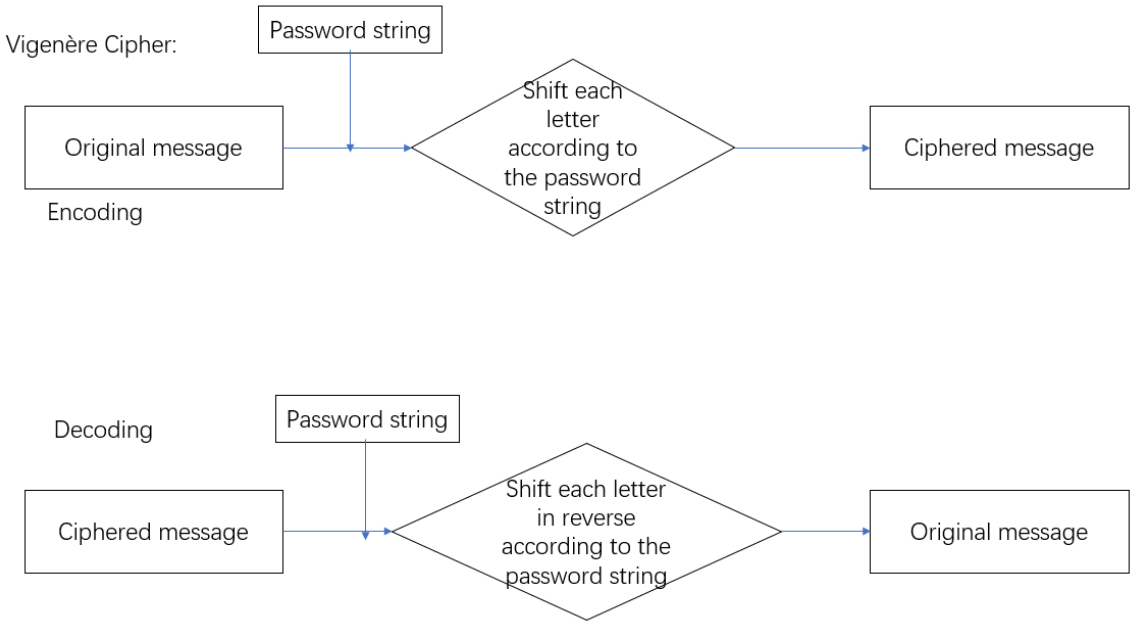
File name: README.pdf
Created By: Yuxiao Wu
Created on: 10/28/2021
No collaborators, no late days
Source: Textbook, wiki

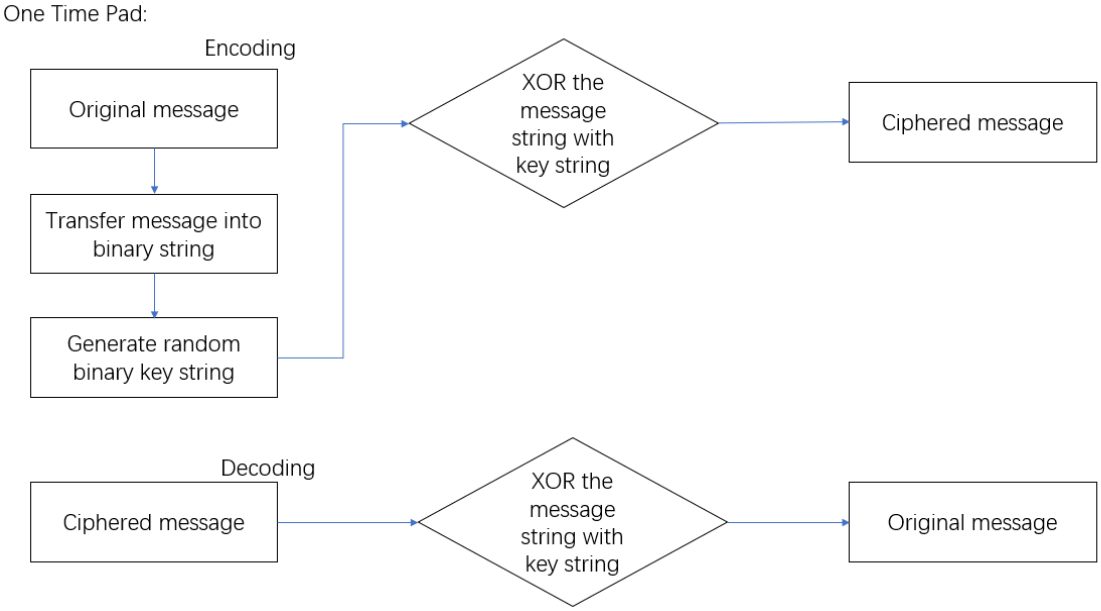
Part 0: Design

In this lab we are going to build three different encrypting methods which are Caesar cipher, Vigenère cipher and one time pad method.





One Time Pad:



Extra Credit

1. Break Vigenère

   See in cipher.py

2. Do some external research and write a paragraph explaining why and how OTP works and briefly illustrate (5-10 sentences) why it's secure (3 points)

   The one-time pad (OTP) is an encryption technique, in this technique, the original message is paired with a random generated key. Then each bit of the original message is encrypted by combining it with the key bit. one-time pad technique cannot be cracked if the four conditions are met:

   1. The key must be truly random.
   2. The key must be at least as long as the plaintext.
   3. The key must never be reused in whole or in part.
   4. The key must be kept completely secret by the communicating parties.

   Different from the Caesar cipher and Vigenère cipher, the rotation for every letter in ciphered message using One time pad is total random. So, it is extremely hard to break the one time pad using the same way breaking Caesar cipher and Vigenère cipher. For example, in order to decrypting a five-letter word, attacker has to try 26^5 which is more than 10 million times in the worst case to find the original message.

3. Do some external research and write a paragraph explaining why reusing keys makes OTP insecure (2 points)

   For example, let p1 and p2 be two original message to send, c1 and c2 be the ciphered message, and k be the common key used for these two messages. Then we have the

math equations: $$c_1 = p_1 \oplus k$$ $$c_2 = p_2 \oplus k$$ we can derive that p1 xor p2 = c1 xor x2. Because the attacker can see both c1 and c2, so the attacker can know the result of p1 xor p2. After that, the attacker can use heuristic cryptanalysis with possibly a few ambiguities to recover the original message by finding the overlaps of the original message.

4. Write a function to attack OPT

   See in cipher,py.