CSI4108 Assignment 3

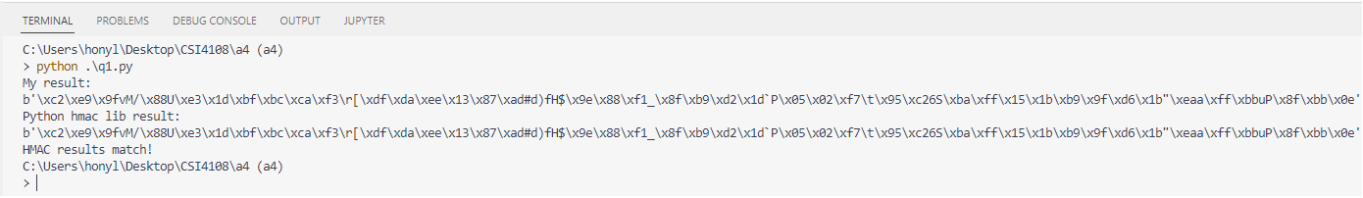
Hongyi Lin, 300053082

General

`math_helper.py` contains the helper functions to do the required math operations.

Question 1

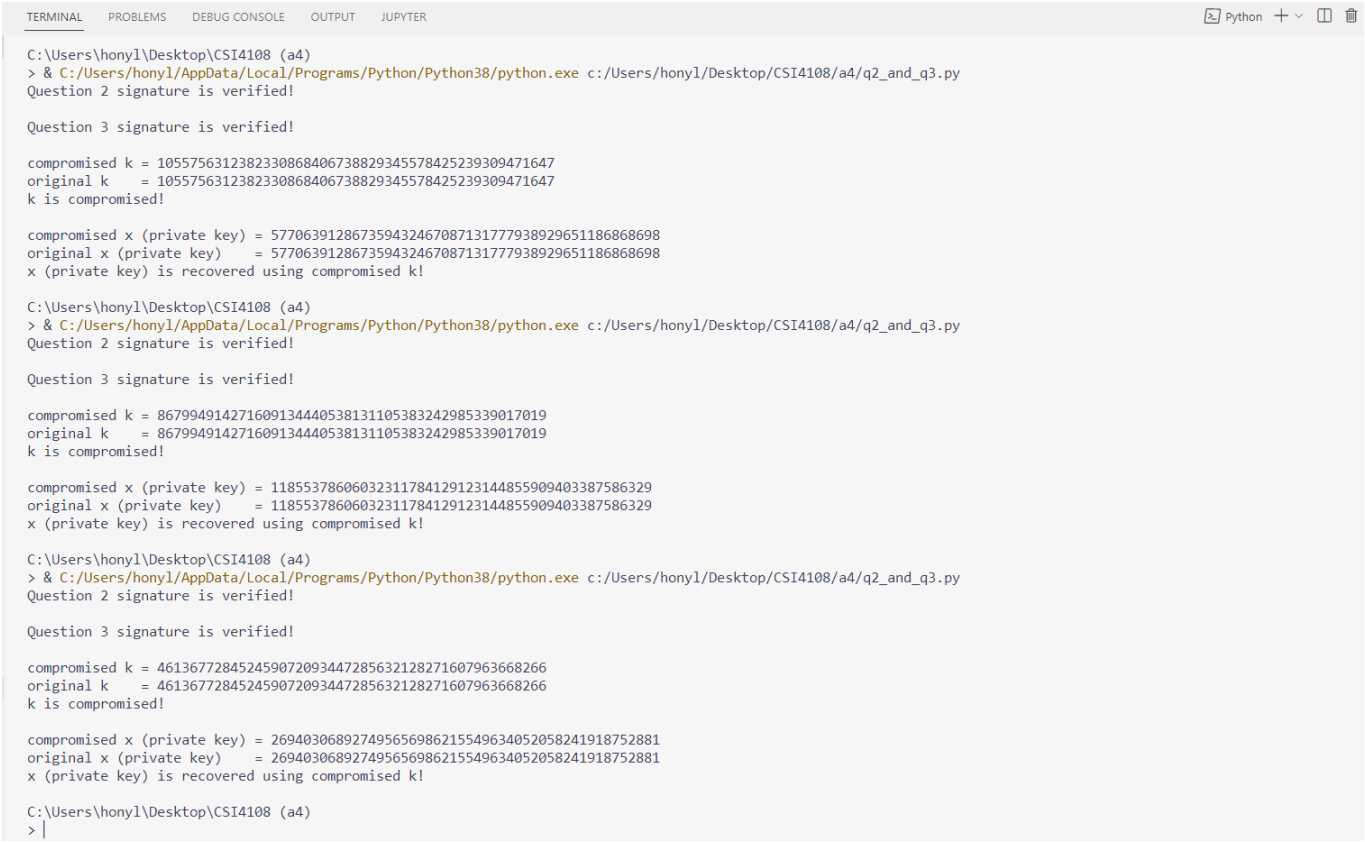
source code: `q1.py`



Question 2 and Question 3

source code:

- `dsa_num_gen.py`: Generate the system parameters of DSA, 1024-bit prime `p`, 160-bit prime `q`, and the generator `g`.
- `dsa_num.py`: Generated files by `dsa_num_gen.py` containing the DSA system parameters.
- `dsa.py`: Contain the `Dsa` class to implement the DSA algorithm with the sign and verify operations. This class use `hashlib.sha1` as the default hash function as required by the question 2.
- `q2_and_q3.py`: the script to run the simulation described in the PDF.



As we can see the results in the screenshot above, the k is compromised each time from the two signatures using the same k. From the compromised k, the private key x can also be recovered each time.

Question 4

Build the block cipher based on the Feistel Netowrk, and use the hash functions instead of the S-boxes in the round function in Feistel Network.

Because of the nice property of the Feistel Network, the hash functions and round function are not required to be invertible. Each time the decryption is needed, just use the round keys in the reverse order to decrypt the ciphertext.