Students are required to build a software package for secure text exchange between two users using  
the Python Cryptography package:  
<https://cryptography.io/en/latest/>

**Part I. Sockets**  
Write a pair of client-server programs that use “sockets” to establish a communication channel between two users (User A and User B). Information on socket programming in Python can be found at: <https://docs.python.org/3/howto/sockets.html>

Make sure that  
1. User A can use the client software to type in a text on a terminal window and, upon pressing the return key, send the text over the sockets to User B who uses the server software to receive the text.  
2. Likewise, User B can type in a text on his terminal and press the return key to send the text over the sockets to User A.  
3. your programs should work not only on the same computer but also across the network: The client and the server software can be run on different computers over a LAN or the Internet.

**Part II. Message encryption & decryption**

Build an encryption program that encrypts a text message using an “authenticated encryption with associated data (AEAD)” mode, and produces a ciphertext as output. Build a second program that,  
given an encrypted file as input, correctly decrypts it back to the original text. This assignment requires students to the Ascon light weight cipher as the AEAD method. A Python library for Ascon AEAD encryption/decryption can be found at:

<https://github.com/meichlseder/pyascon/tree/5ee786cdc8a74d9c0f7b3c81f99f5dcb5490ca00>

The general form of AEAD encryption using Ascon is  
(𝐶, 𝑇) = 𝑎𝑠𝑐𝑜𝑛\_𝑒𝑛𝑐𝑟𝑦𝑝𝑡(𝑘𝑒𝑦, 𝑛𝑜𝑛𝑐𝑒, 𝑎𝑠𝑠𝑜𝑐𝑖𝑎𝑡𝑒𝑑𝑑𝑎𝑡𝑎, 𝑝𝑙𝑎𝑖𝑛𝑡𝑒𝑥𝑡, 𝑣𝑎𝑟𝑖𝑎𝑛𝑡 = "𝐴𝑠𝑐𝑜𝑛 − 128"), where 𝑝𝑙𝑎𝑖𝑛𝑡𝑒𝑥𝑡 represents a text message. Note that the output contains 2 parts: 𝐶 is the ciphertext produced by symmetric encryption, and 𝑇 is an authentication tag. Likewise, the general form of AEAD decryption is 𝑀 = 𝑎𝑠𝑐𝑜𝑛\_𝑑𝑒𝑐𝑟𝑦𝑝𝑡(𝑘𝑒𝑦, 𝑛𝑜𝑛𝑐𝑒, 𝑎𝑠𝑠𝑜𝑐𝑖𝑎𝑡𝑒𝑑𝑑𝑎𝑡𝑎, 𝑐𝑖𝑝ℎ𝑒𝑟𝑡𝑒𝑥𝑡, 𝑣𝑎𝑟𝑖𝑎𝑛𝑡 = "𝐴𝑠𝑐𝑜𝑛 − 128"),  
Note that 𝑐𝑖𝑝ℎ𝑒𝑟𝑡𝑒𝑥𝑡 = (𝐶, 𝑇) . Also note that in addition to the 𝑘𝑒𝑦, both encryption and decryption must use the same 𝑛𝑜𝑛𝑐𝑒 and the same 𝑎𝑠𝑠𝑜𝑐𝑖𝑎𝑡𝑒𝑑𝑑𝑎𝑡𝑎.

Decide ahead of time the following 3 data items (𝑘𝑒𝑦, 𝑛𝑜𝑛𝑐𝑒, 𝑎𝑠𝑠𝑜𝑐𝑖𝑎𝑡𝑒𝑑𝑑𝑎𝑡𝑎) to be used by both the client and the server.  
• A shared secret key 𝑘𝑒𝑦 for encrypting and decrypting a text message. It is typically generated uniformly at random or derived from a secure & random seed. It must be kept secret. The length of 𝑘 is dependent on the mode to be used.  
• A 𝑛𝑜𝑛𝑐𝑒. For this assignment, let’s set 𝑛𝑜𝑛𝑐𝑒 to 0. It does not have to be kept secret.  
• Associated data (to indicate the context). It does not have to be kept secret. For this assignment, let’s use 𝑎𝑠𝑠𝑜𝑐𝑖𝑎𝑡𝑒𝑑𝑑𝑎𝑡𝑎=“CS645/745 Modern Cryptography: Secure Messaging”.  
IMPORTANT:  
Do NOT use Python Fernet (https://cryptography.io/en/latest/fernet/ ) for this assignment!