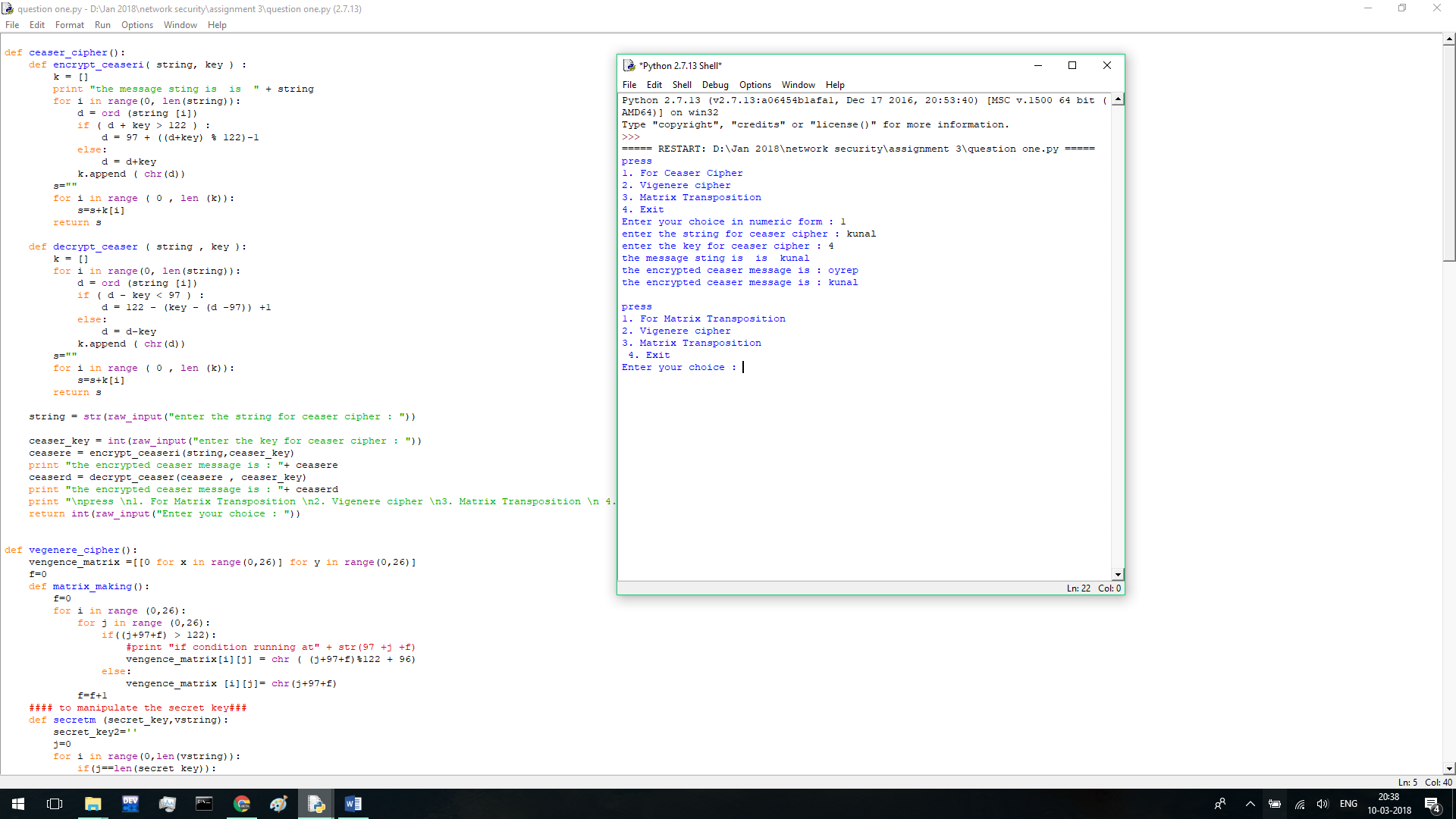
**Network Security Assignment 3**

**Kunal Kathpal**

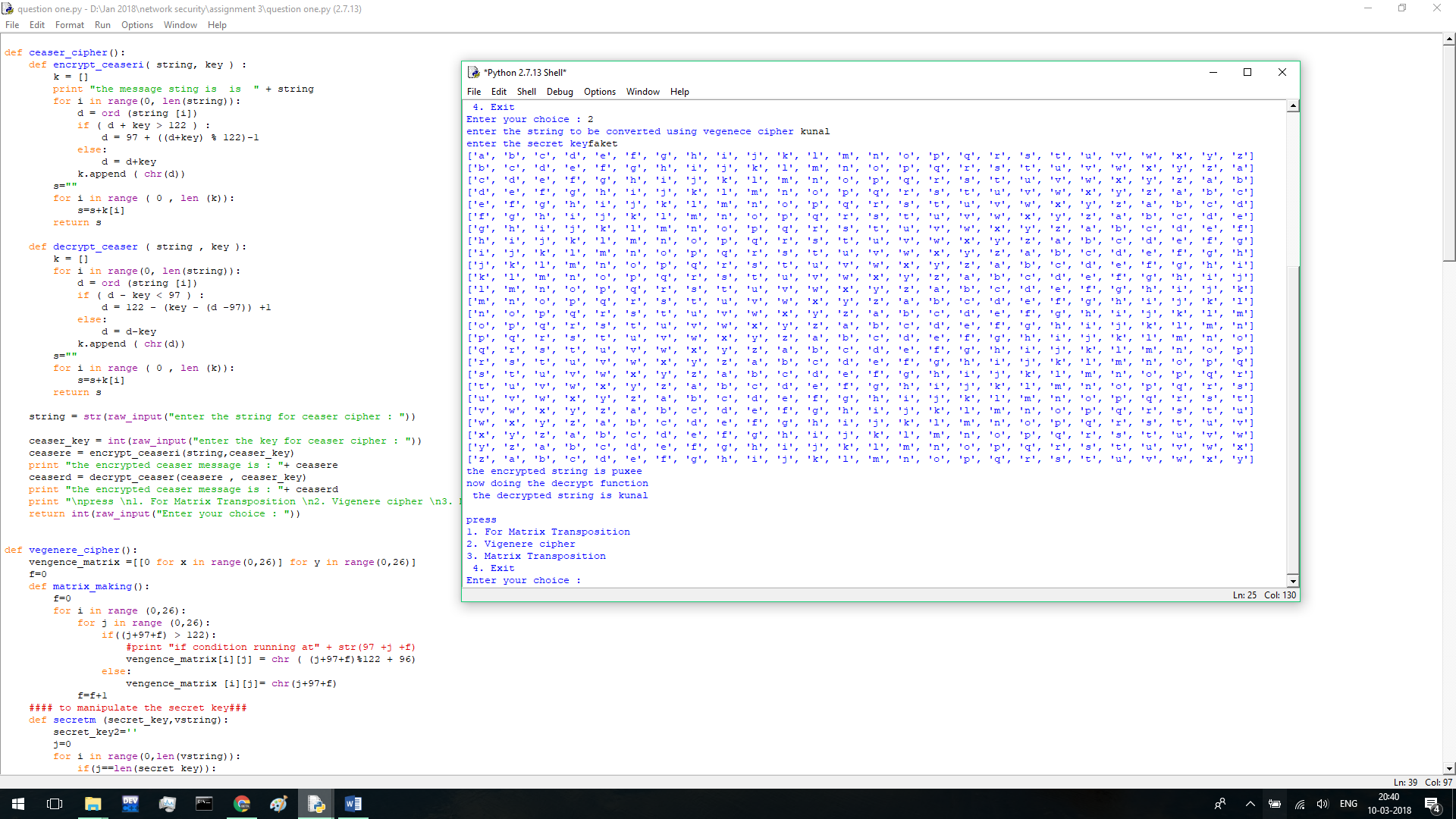
**B00765934**

**QUESTION 1**

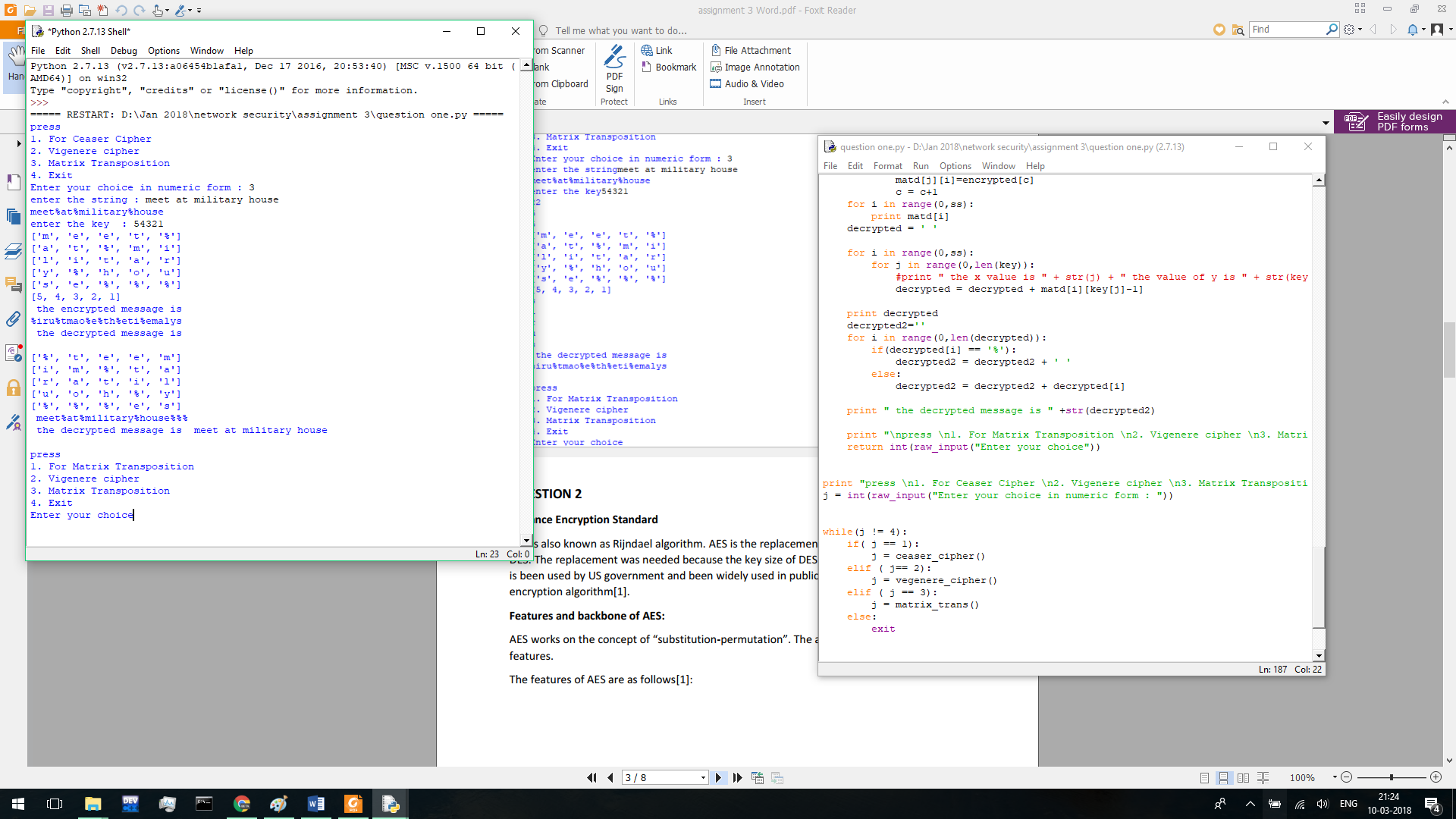
**1.1**



**1.2**



**1.3**



**QUESTION 2**

**Advance Encryption Standard**

AES is also known as Rijndael algorithm. AES is the replacement of the famous algorithm DES. The replacement was needed because the key size of DES was too small. The algorithm is been used by US government and been widely used in public too as it is a popular encryption algorithm[1].

**Features and backbone of AES:**

AES works on the concept of “substitution-permutation”. The algorithm has following features.

The features of AES are as follows[1]:

* Symmetric key symmetric block cipher
* 128 – bit data but the key size can be different and ranges from 128/192/256-bit keys
* Stronger and faster than Triple-DES
* Provide full specification and design details.

**Operation of AES**

AES is an iterative rather than Feistel cipher. It is based on ‘substitution–permutation network’. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations)[2].

Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix[2] –

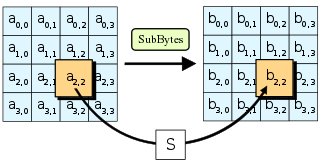


Figure 1 :In the SubBytes step, each byte in the state is replaced with its entry in a fixed 8-bit lookup table, S; bij = S(aij). [2]

Unlike DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

Figure 2: The figure shows hoe the cipher key is converted into rounds key of 128 bits each key is given as input to each round of encryption process.

**Key generation in AES**

Rijndael’s key schedule is used to expand single key into a separate round keys. The round keys are produced by the key schedule from the initial key. It does the certain operations – Rotate, Rcon, S-Box and Key schedule score. Here S-Box is a matrix which serves a lookup table [1].

**Encryption Process**

The image below will show the Encryption Process in AES.



There are three subprocess in Encryption Process.

1. Shiftrows[1]

Each of the four rows of the matrix is shifted to the left. Any entries that ‘fall off’ are re-inserted on the right side of row. Shift is carried out as follows −

First row is not shifted.

Second row is shifted one (byte) position to the left.

Third row is shifted two positions to the left.

Fourth row is shifted three positions to the left.

The result is a new matrix consisting of the same 16 bytes but shifted with respect to each other.

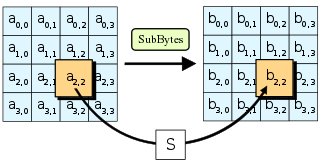


Figure 3 : In the SubBytes step, each byte in the state is replaced with its entry in a fixed 8-bit lookup table, S; bij = S(aij) [2].

1. Mix Columns[1]

Each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round[1].

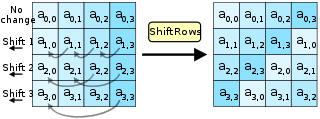


Figure 4 : In the ShiftRows step, bytes in each row of the state are shifted cyclically to the left. The number of places each byte is shifted differs for each row.[2]

1. The Mix Columns Step[1]

n the MixColumns step, the four bytes of each column of the state are combined using an invertible [linear transformation](https://en.wikipedia.org/wiki/Linear_transformation). The MixColumnsfunction takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes. Together with ShiftRows, MixColumns provides [diffusion](https://en.wikipedia.org/wiki/Diffusion_(cryptography)) in the cipher[2].

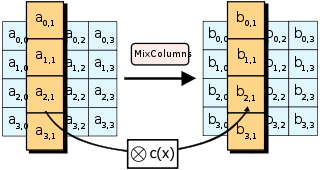


Figure 5 : In the MixColumns step, each column of the state is multiplied with a fixed polynomial {\displaystyle c(x)} c(x). [2]

1. Add Round Keys[2]

The matrix of 16 bytes are been xored with the 128 bit size key. If this is the last round the output is considered to be a s the cipher text.

If this is not the last step they are been considered as 16 bytes and again the process continues.

**Decryption Process**

In decryption process the AES ciphertext is similar to the encryption but its in reverse order. Here it’s like[1]

1. Add round key
2. Mix Column
3. Shift Rows
4. Byte Substitution

**How secure is AES:**

Aes uses a long key for encryption as a result its more secure that DES. The key combination can be upto 2^32 that’s somewhat upto 4 Billion. The AES algorithm is considered very secure algorithm for encryption. The National Security Agency announced in June 2003 that AES can be used to protect classified information [1]. In June 2009, there was an attack on AES on its specific implementation [1]. It did not attack the cipher, but on its implementation which inadvertently leak information. In April 2005, D.J. Bernstein used a cache-timing attack that he made on a custom server that used AES encryption [1].

References:

[1] <https://en.wikipedia.org/wiki/Advanced_Encryption_Standard>

[2] <https://www.tutorialspoint.com/cryptography/advanced_encryption_standard.htm>

**Question 3**

