**Advanced Encryption Standard(AES)**

**CSE 459: Cryptography & Network Security**

Submitted by

Name : Pranav Mudigonda

Roll No: AP22110010765

Section: CSE Y

Lab Date:03/02/2025

Submission Date: 09/02/2025

**A picture containing text

Description automatically generated**

**Department Computer Science and Engineering**

**School of Engineering and Sciences**

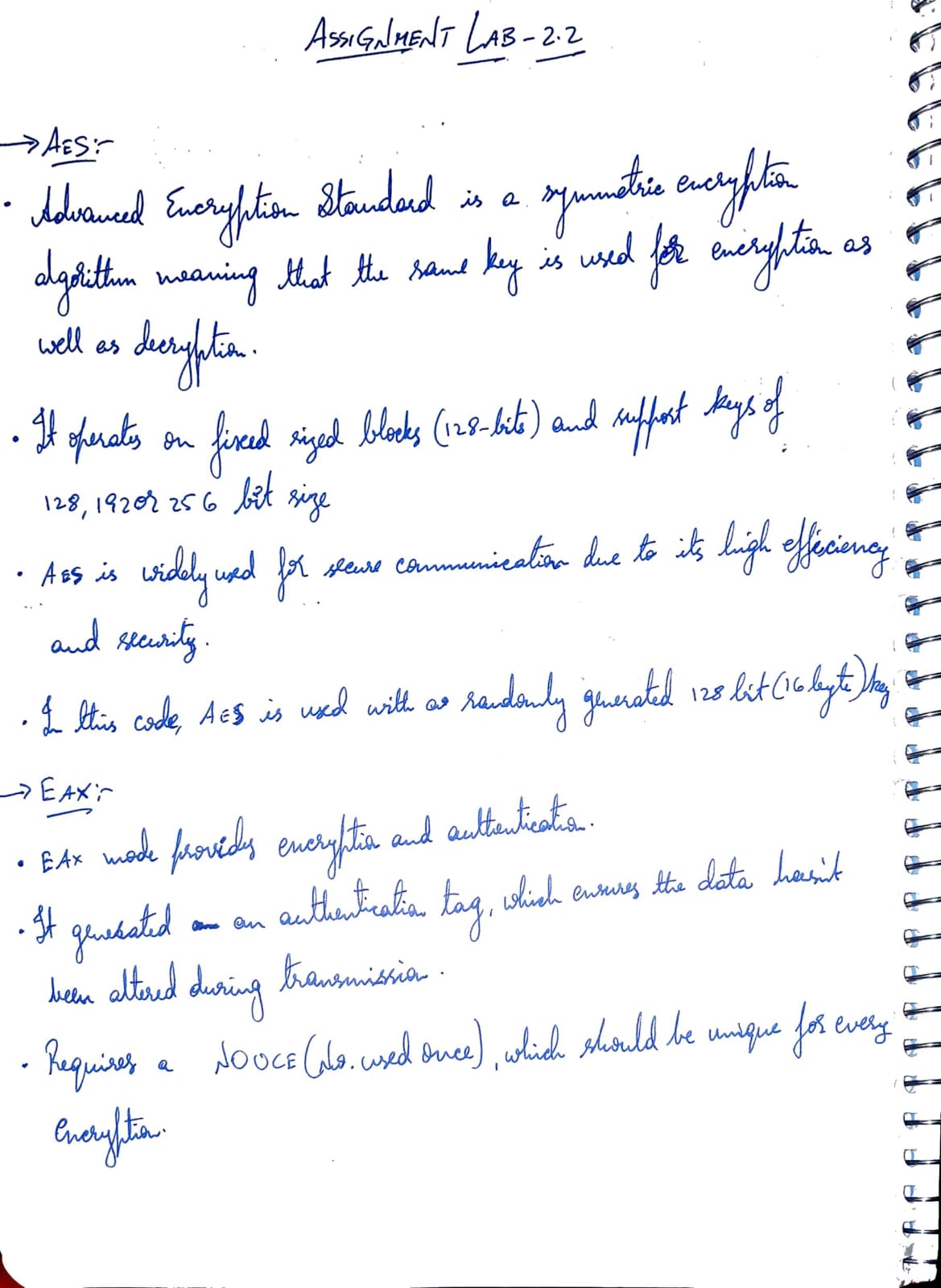
**SRM University–AP**

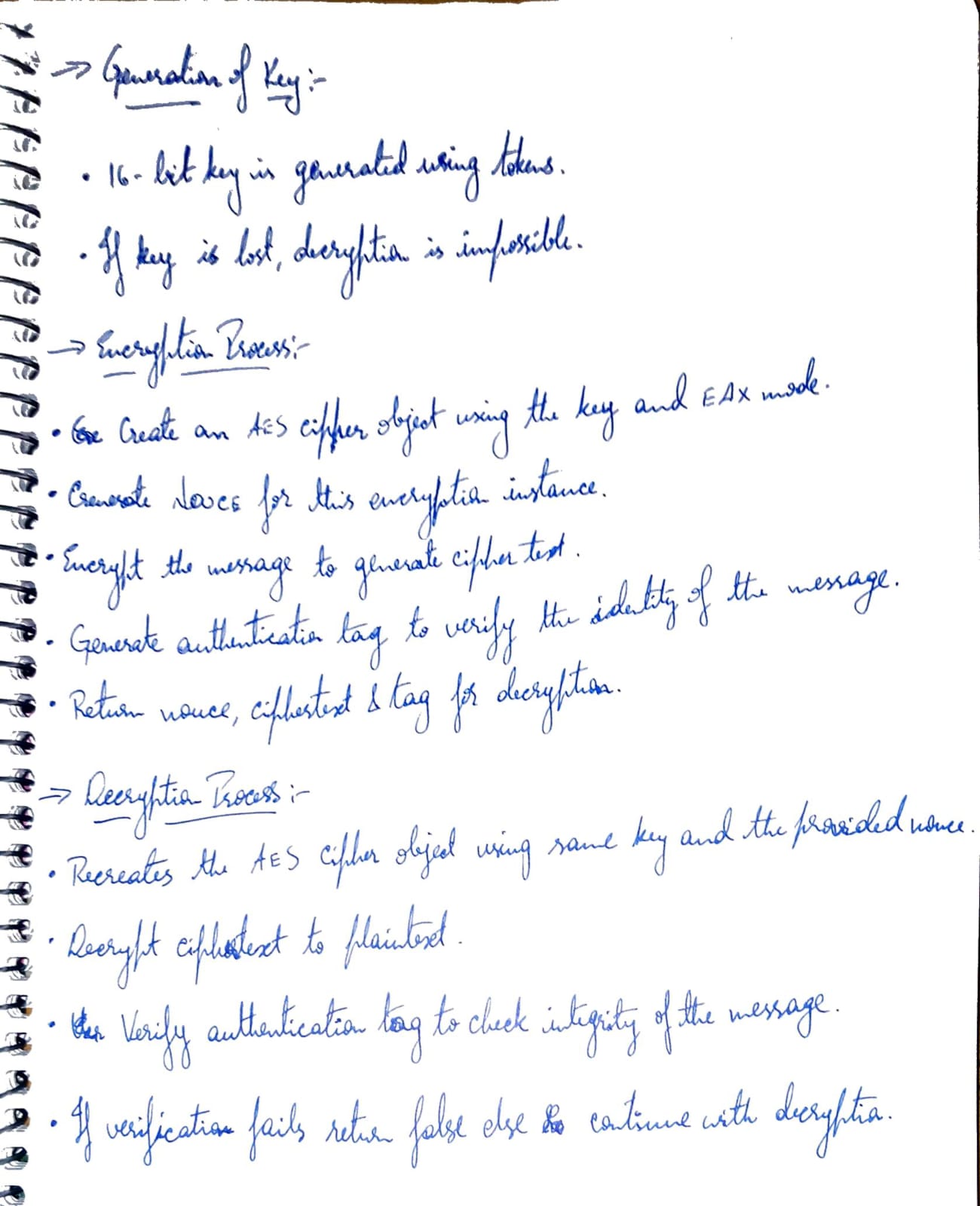
**Amaravati, Andhra Pradesh – 522240, India**

**Question**

**1) Implement AES using the available library to encrypt and decrypt a given plain text file.**

**2) Extend the program (2) to encrypt and decrypt a given file.**

**Algorithm Description:**



**Solution :**

from Crypto.Cipher import AES

from secrets import token\_bytes

key = token\_bytes(16)

def aes\_encrypt(msg):

cipher = AES.new(key, AES.MODE\_EAX)

nonce = cipher.nonce

ciphertext, tag = cipher.encrypt\_and\_digest(msg.encode('ascii'))

return nonce, ciphertext, tag

def aes\_decrypt(nonce, ciphertext, tag):

cipher = AES.new(key, AES.MODE\_EAX, nonce=nonce)

plaintext = cipher.decrypt(ciphertext)

try:

cipher.verify(tag)

return plaintext.decode('ascii')

except:

return False

nonce, ciphertext, tag = aes\_encrypt(input("Enter a message: "))

plaintext = aes\_decrypt(nonce, ciphertext, tag)

print(f'\n Plain Text Entered : {plaintext} \n Cipher Text in base 64 : {ciphertext} , \n nonce : {nonce} \n tag : {tag}\n')

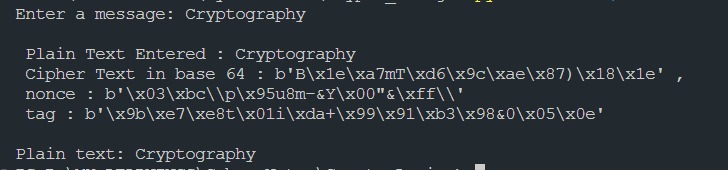
if not plaintext:

print("Message is corrupted")

else:

print(f'Plain text: {plaintext}')

**Output:**

****

**Code Repository:**

<https://github.com/pranav2885/crypto-lab>