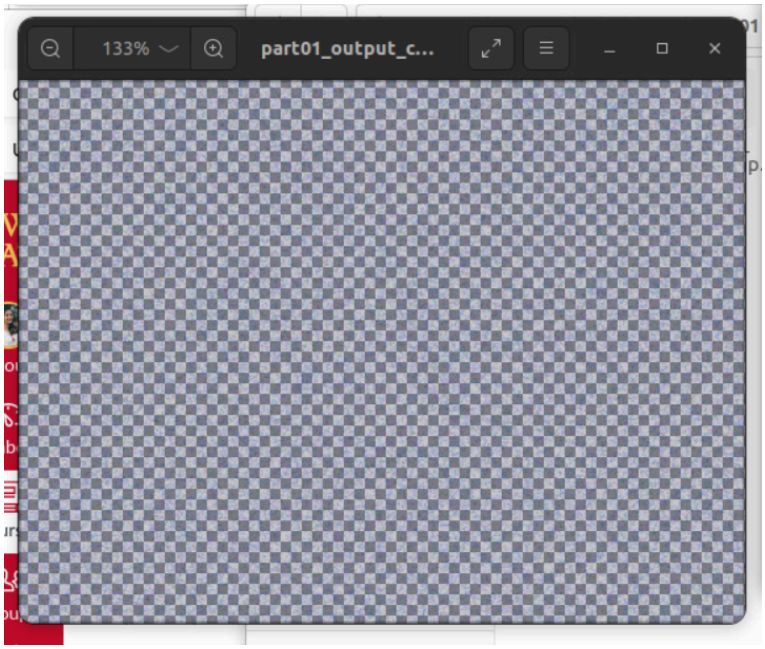
Lab 07 Template

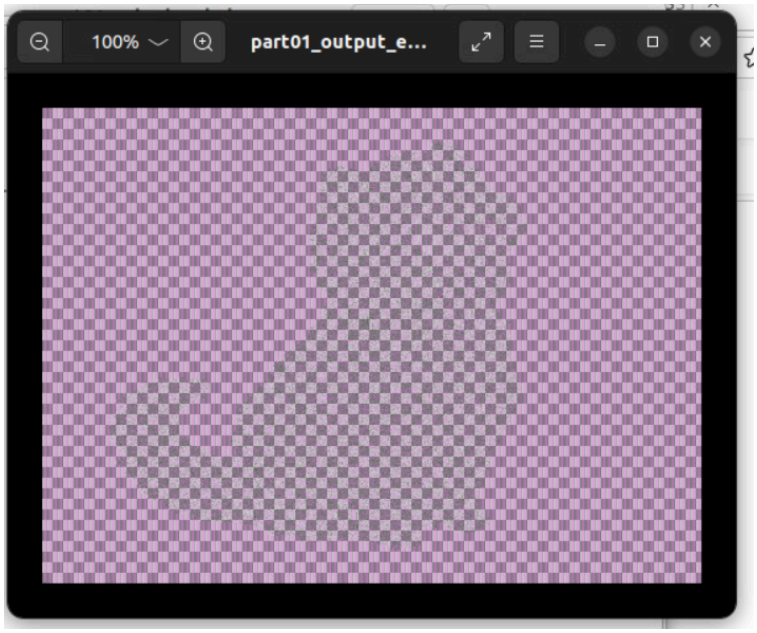
Part 01

1.

Open the two images and take a screenshot of them.

(10 points)





2.

How does ECB mode handle these repetitive blocks, and why does this lead to

visible patterns in the encrypted image?

(20 points total, 10 points each question)

ECB mode encrypts identical plaintext blocks into identical ciphertext blocks independently. This preserves structural patterns in images. ECB does not mix plaintext data between blocks or use vectors allowing attackers to recognize encrypted structures and perform frequency analysis on ciphertext patterns. The visible patterns emerge because the viewers can still render the encrypted BMP header while displaying ciphertext blocks that retain original spatial relationships. Identical input blocks always produce identical output blocks.

3.

What effect does CBC mode have on repetitive structures in the image compared

to ECB mode? How does the introduction of the Initialization Vector (IV) in CBC

mode help in obscuring patterns that are visible in ECB mode?

(20 points total, 10 points each question)

CBC mode eliminates pattern visibility in ECB by introducing chained encryption and initialization vector. Unlike ECB's independent block processing, CBC XOR s each plaintext block with previous ciphertext block before encryption, creating dependency chains between blocks. IV ensures unique encryption results even for identical first blocks, while subsequent blocks inherit variability through this chaining process. IV breaks initial pattern alignment and chaining propagates randomness through all subsequent blocks. This effectively converts structured image data into ciphertext that resembles noise unlike ECB.

4.

Upload python code for part 01.

(10 points)

Uploaded.

Part 02

1.

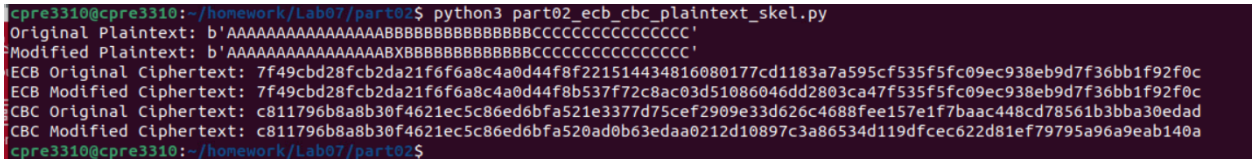
Please take a screenshot of the original and modified plaintext. (5 points)

```
24      # Original plaintext
25      plaintext = b'AAAAAAAAAAAAAAAABBBBBBBBBBBBBBBBCCCCCCCCCCCCCCCC'
26      #TODO: You are expected to change one byte of the modified_plaintext. You
    should
27      #       change a character in the second block of the plaintext. Note that,
28      #       each character is 1 byte, and the block size is 16 bytes.
29      modified_plaintext = b'AAAAAAAAAAAAAAAAAXBBBBBBBBBBBBBBBCCCCCCCCCCCCCCCC'
30      # Plaintext with a single bit change (flipping the last bit of the last byte)
31
```

2.

Please take a screenshot of the ECB Original Ciphertext, ECB Modified Ciphertext,

CBC Original Ciphertext, and CBC Modified Ciphertext. (5 points)



3. What do you observe in the ciphertexts when you compare ECB original and

modified ones? (10 points)

Only the second half of the ciphertext changes. The first and third blocks remain identical

original ciphertext. The change is localized to the block where the plaintext was modified.

4. What do you observe in the ciphertexts when you compare CBC original and

modified ones? What is the difference in terms of changed/unchanged parts of

ECB and CBC modified ciphertexts? (10 points)

The first block remains unchanged. The second block and all subsequent blocks are completely different from the original ciphertext. Change in one block propagates affecting all following blocks.

The key difference is that in ECB, changes are isolated to the modified block, while in CBC changes propagate through the entire ciphertext from the point of modification onward. This shows CBC's superior ability to obscure patterns in the plaintext.

5.

How would this observation affect the decrypted plaintext after the modifications?

(This would be related to the Exam-1 question as well.) (10 points)

For ECB mode only the second block of the decrypted plaintext would be different from the original. For CBC mode the first block of the decrypted plaintext would remain unchanged. The key different in impact is that in ECB the modification affects only the corresponding block in decrypted plaintext while in CBC the modification causes an avalanche effect changing all subsequent blocks in the decrypted plaintext after the point of change.