COMSAT UNIVERSITY ISLAMABAD ATTOCK CAMPUS

LAB MID INFORMATION SECURITY

NAME: MENAHIL NOOR

REGISTRATION NO: SP24-BSE-035

SUBMITTED TO: MA'AM AMBREEN GULL

DEPARTMENT: SOFTWARE ENGINEERING

DATE: 21 OCTOBER 2025

QUESTION: 02

Caesar Cipher (Decryption) [10 Marks]

Write a Python program to decrypt a message that was encrypted using the Caesar Cipher. The program should take ciphertext (LXFOPVEFRNHR) and key (5) as input and display the plaintext.
Example:
Enter ciphertext: khoor
Enter shift: 3
Plaintext: hello
Hint: Use ord() and chr() for letter shifting backward.

ANSWERE

CODE:

caesar_decrypt.py × vigenere_decrypt.py caesar_encrypt_fixed.py

D: > PyCharm 2025.2.1.1 > plugins > python-ce > helpers > typeshed > stubs > hvac > hvac > api > auth_methods > caesar_decrypt.py > ...

```python

ciphertext = input("Enter ciphertext: ")
shift = int(input("Enter shift: "))

plaintext = ""


for char in ciphertext:
    if char.isalpha():
        if char.isupper():
            plaintext += chr((ord(char) - ord('A') - shift) % 26 + ord('A'))
        else:
            plaintext += chr((ord(char) - ord('a') - shift) % 26 + ord('a'))
    else:
        plaintext += char


print("Plaintext:", plaintext)

```

OUTPUT:

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

PS C:\Users\Discount Laptop> & "C:/Users/Discount Laptop/AppData/Local/Programs/Python/Python313/python.exe" "d:/PyCharm 2025.2.1.1/plugins/py
hvac/api/auth_methods/caesar_decrypt.py"
Enter ciphertext: khoor
Enter shift: 3
Plaintext: hello
PS C:\Users\Discount Laptop> & "C:/Users/Discount Laptop/AppData/Local/Programs/Python/Python313/python.exe" "d:/PyCharm 2025.2.1.1/plugins/py
hvac/api/auth_methods/caesar_decrypt.py"
Enter ciphertext: MANAHIL
Enter shift: 4
Plaintext: IWJWDEH
PS C:\Users\Discount Laptop>
```

QUESTION: 03

Question 3 – Vigenère Cipher (Decryption Only) [5 Marks]

Write a Python program to decrypt a ciphertext using the Vigenère Cipher. Ask the user for ciphertext and key, and display the decrypted plaintext.

Example:
Enter ciphertext: LXFOPVEFRNHR
Enter key: LEMON
Plaintext: ATTACKATDAWN

ANSWERE

CODE:

caesar_decrypt.py | vigenere_decrypt.py | caesar_encrypt_fixed.py

D: > PyCharm 2025.2.1.1 > plugins > python-ce > helpers > typeshed > stubs > hvac > hvac > api > auth_methods > vigenere_decrypt.py > ...

```python

ciphertext = input("Enter ciphertext: ").upper()
key = input("Enter key: ").upper()

plaintext = ""
key_index = 0

for char in ciphertext:
    if char.isalpha():

        shift = ord(key[key_index]) - ord('A')


        decrypted_char = chr((ord(char) - ord('A') - shift) % 26 + ord('A'))
        plaintext += decrypted_char


        key_index = (key_index + 1) % len(key)
    else:
        plaintext += char


print("Plaintext:", plaintext)

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Enter key: LEMON
Plaintext: ATTACKATDAWN
PS C:\Users\Discount Laptop>
PS C:\Users\Discount Laptop> & "C:/Users/Discount Laptop/AppData/Local/Programs/Python/Python313/python.exe" "d:/PyCharm 2025.2.1.1/plugins/pytho
hvac/api/auth_methods/vigenere_decrypt.py"
Enter ciphertext: MANAHIL
Enter key: NOOR
Plaintext: ZMZJUUX
PS C:\Users\Discount Laptop>
```

OUTPUT:

QUESTION: 04

Question 4 – Debugging Task (Caesar Cipher Code) [5 Marks]

The following program is intended to encrypt text using the Caesar Cipher, but it contains an error. Fix the mistake so that it runs correctly and gives the right output.

- ```
  def caesar_encrypt(text, shift):
      result = ""
      for char in text:
          if char.isalpha():
              result += chr(ord(char) + shift)
          else:
              result += char
      return result

  msg = input("Enter message: ")
  s = int(input("Enter shift: "))
  print("Ciphertext:", caesar_encrypt(msg, s))
  ```

Hint: The code doesn't wrap around alphabets (A–Z or a–z). Use modular arithmetic to fix the shifting logic.

ANSWERE

CODE:
  ```

caesar_decrypt.py | vigenere_decrypt.py | caesar_encrypt_fixed.py

D: > PyCharm 2025.2.1.1 > plugins > python-ce > helpers > typeshed > stubs > hvac > hvac > api > auth_methods > caesar_encrypt_fixed.py > ...

```python

def caesar_encrypt(text, shift):
    result = ""
    for char in text:
        if char.isalpha():
            if char.isupper():
                result += chr((ord(char) - ord('A') + shift) % 26 + ord('A'))
            else:
                result += chr((ord(char) - ord('a') + shift) % 26 + ord('a'))
        else:
            result += char
    return result

msg = input("Enter message: ")
s = int(input("Enter shift: "))
print("Ciphertext:", caesar_encrypt(msg, s))

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Discount Laptop> & "C:/Users/Discount Laptop/AppData/Local/Programs/Python/Python313/python.exe" "d:/PyCharm 2025.2.1.1/plugins/python-ce/h
hvac/api/auth_methods/caesar_encrypt_fixed.py"
Enter message: hello
Enter shift: 3
Ciphertext: khoor
PS C:\Users\Discount Laptop> & "C:/Users/Discount Laptop/AppData/Local/Programs/Python/Python313/python.exe" "d:/PyCharm 2025.2.1.1/plugins/python-ce/h
hvac/api/auth_methods/caesar_encrypt_fixed.py"
Enter message: noor
Enter shift: 4
Ciphertext: rssv
PS C:\Users\Discount Laptop>
```

OUTPUT:

QUESTION: 05

Question 5 – Conceptual: DES and AES **[5 Marks]**

Answer briefly:

a) Write one similarity between DES and AES.

Both **DES (Data Encryption Standard)** and **AES (Advanced Encryption Standard)** are **symmetric key block ciphers**, meaning they use the **same key** for both **encryption and decryption**.

b) What does CBC mode stand for in block ciphers?

CBC stands for **Cipher Block Chaining** mode — in this mode, each plaintext block is **XORed with the previous ciphertext block** before encryption, providing stronger security by making each block depend on the previous one.

c) Why is AES faster than DES?

AES is faster because it **uses fewer and more efficient rounds (based on byte-level operations)** and supports **larger block sizes (128 bits)**, whereas DES uses **smaller 64-bit blocks** and more complex bit-level operations.