

```

In [ ]: def caesar_cipher(text, shift):
    encrypted_text = ""
    for char in text:
        if char.isalpha():
            # Xác định vị trí của ký tự trong bảng chữ cái
            is_upper = char.isupper()
            char = char.lower()
            char_code = ord(char) - ord('a')

            # Áp dụng dịch chuyển
            char_code = (char_code + shift) % 26

            # Chuyển lại thành ký tự và giữ nguyên chữ hoa nếu cần
            char_code += ord('a')
            if is_upper:
                char_code -= 32 # Chuyển thành chữ hoa nếu ký tự gốc là chữ hoa

            encrypted_text += chr(char_code)
        else:
            # Giữ nguyên ký tự không phải chữ cái
            encrypted_text += char

    return encrypted_text

def caesar_decipher(text, shift):
    # Sử dụng hàm mã hóa với số dịch chuyển âm để giải mã
    return caesar_cipher(text, -shift)

# Ví dụ mã hóa
plaintext = "Hello, World!"
shift_amount = 3
encrypted_text = caesar_cipher(plaintext, shift_amount)
print("Mã hóa:", encrypted_text)

# Ví dụ giải mã
decrypted_text = caesar_decipher(encrypted_text, shift_amount)
print("Giải mã:", decrypted_text)

```

```

In [ ]: def vigenere_cipher(text, key):
    encrypted_text = ""
    key_length = len(key)
    for i in range(len(text)):
        char = text[i]
        if char.isalpha():
            is_upper = char.isupper()
            char = char.lower()
            key_char = key[i % key_length].lower()

            # Xác định sự dịch chuyển cần thiết
            shift = ord(key_char) - ord('a')
            char_code = ord(char) + shift

            # Đảm bảo char_code nằm trong khoảng chữ cái a-z
            if char.islower() and char_code > ord('z'):

```

```

        char_code -= 26
    elif char.isupper() and char_code > ord('Z'):
        char_code -= 26

    encrypted_text += chr(char_code).upper() if is_upper else chr(char_code)
else:
    encrypted_text += char

return encrypted_text

def vigenere_decipher(text, key):
    decrypted_text = ""
    key_length = len(key)
    for i in range(len(text)):
        char = text[i]
        if char.isalpha():
            is_upper = char.isupper()
            char = char.lower()
            key_char = key[i % key_length].lower()

            # Xác định sự dịch chuyển cần thiết
            shift = ord(key_char) - ord('a')
            char_code = ord(char) - shift

            # Đảm bảo char_code nằm trong khoảng chữ cái a-z
            if char.islower() and char_code < ord('a'):
                char_code += 26
            elif char.isupper() and char_code < ord('A'):
                char_code += 26

            decrypted_text += chr(char_code).upper() if is_upper else chr(char_code)
        else:
            decrypted_text += char

    return decrypted_text

# Ví dụ mã hóa và giải mã
plaintext = "Hello, World!"
key = "KEY"
encrypted_text = vigenere_cipher(plaintext, key)
print("Mã hóa:", encrypted_text)

decrypted_text = vigenere_decipher(encrypted_text, key)
print("Giải mã:", decrypted_text)

```