Code:

import numpy as np

from string import ascii\_uppercase

def text\_to\_numbers(text):

    return [ascii\_uppercase.index(c) for c in text]

def numbers\_to\_text(numbers):

    return ''.join(ascii\_uppercase[n % 26] for n in numbers)

def prepare\_text(text, size):

    text = text.upper().replace(" ", "")

    while len(text) % size:

        text += "X"

    return text

def encrypt(plain, key):

    size = len(key)

    plain = prepare\_text(plain, size)

    key\_matrix = np.array(key)

    cipher\_text = ""

    for i in range(0, len(plain), size):

        block = np.array(text\_to\_numbers(plain[i:i+size]))

        cipher\_block = np.dot(key\_matrix, block) % 26

        cipher\_text += numbers\_to\_text(cipher\_block)

    return cipher\_text

def decrypt(cipher, key):

    size = len(key)

    key\_matrix = np.array(key)

    inv\_key = np.linalg.inv(key\_matrix) \* np.linalg.det(key\_matrix)

    inv\_key = np.round(inv\_key \* pow(int(np.linalg.det(key\_matrix)), -1, 26)) % 26

    inv\_key = inv\_key.astype(int)

    plain\_text = ""

    for i in range(0, len(cipher), size):

        block = np.array(text\_to\_numbers(cipher[i:i+size]))

        plain\_block = np.dot(inv\_key, block) % 26

        plain\_text += numbers\_to\_text(plain\_block)

    return plain\_text

key = [[6, 24, 1], [13, 16, 10], [20, 17, 15]]

plain\_text = "HELLO"

cipher\_text = encrypt(plain\_text, key)

decrypted\_text = decrypt(cipher\_text, key)

print("Cipher:", cipher\_text)

print("Decrypted:", decrypted\_text)

Output:

Cipher: TFJJZX

Decrypted: HELLOX