program for Caesar cipher, known as the affine Caesar cipher, has the following form: For each plaintext letter p, substitute the ciphertext letter C: C = E([a, b], p) = (ap + b) mod 26 A basic requirement of any encryption algorithm is that it be one-to-one. That is, if p q, then E(k, p) E(k, q). Otherwise, decryption is impossible, because more than one plaintext character maps into the same ciphertext character. The affine Caesar cipher is not one-to-one for all values of a. For example, for a = 2 and b = 3, then E([a, b], 0) = E([a, b], 13) = 3.

def affine\_encrypt(msg,a,b):

return ''.join(chr(((a\*(ord(c)-65)+b)%26)+65) for c in msg.upper())

def affine\_decrypt(cipher,a,b):

a\_inv=pow(a,-1,26)

return ''.join(chr(((a\_inv\*((ord(c)-65)-b))%26)+65) for c in cipher)

msg="HELLO"

a,b=5,8

cipher=affine\_encrypt(msg,a,b)

plain=affine\_decrypt(cipher,a,b)

print("Message:",msg)

print("Cipher:",cipher)

print("Decrypted:",plain)

