CSA5122-CRYPTOGRAPHY FOR NETWORK AND SECURITY

LAB PROGRAMS EXECUTION

13.PUBLIC KEY CRYPTO SYSTEM

import random

def is\_prime(n): return all(n % i for i in range(2, int(n\*\*0.5)+1)) and n > 1

def gen\_prime(): return next(p for p in iter(lambda: random.randint(100, 200), None) if is\_prime(p))

def gcd(a, b): return a if not b else gcd(b, a % b)

def modinv(a, m):

m0, x0, x1 = m, 0, 1

while a > 1: q, a, m, x0, x1 = a // m, m, a % m, x1 - (a // m) \* x0, x0

return x1 + m0 if x1 < 0 else x1

def gen\_keys():

p, q = gen\_prime(), gen\_prime()

while q == p: q = gen\_prime()

n, phi = p \* q, (p - 1) \* (q - 1)

e = next(i for i in range(3, phi, 2) if gcd(i, phi) == 1)

d = modinv(e, phi)

return (e, n), (d, n)

def encrypt(msg, pub): return [(ord(c) \*\* pub[0]) % pub[1] for c in msg]

def decrypt(cipher, priv): return ''.join(chr((c \*\* priv[0]) % priv[1]) for c in cipher)

pub, priv = gen\_keys()

msg = "HELLO"

cipher = encrypt(msg, pub)

print("Encrypted:", cipher)

print("Decrypted:", decrypt(cipher,priv))

