Name:-Amrit Ranjan

Reg No:-21BCE1847

Cryptography And Network Security

Assignment 3

Faculty Name:-Dr Tapabarta Roy

RSA DIFFIE HELMAN



Problem Statement

Choose two distinct primes each consisting of at least three digits and construct a RSA model with clear distinct encryption key and decryption key. Take the bigger prime among the ones that you chose and construct a diffie helman key exchange protocol with suitable parameters.

Code:-

pip install sympy

from sympy import randprime

from sympy import mod\_inverse

import random

import math

# Choose two distinct primes each consisting of at least three digits

p = randprime(100, 999)

q = randprime(1000, 9999)

# Calculate n and phi

n = p \* q

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

# Choose e such that 1 < e < phi and e is co-prime with n and phi

def find\_e(phi):

    while True:

        e = random.randrange(2, phi)

        if math.gcd(e, phi) == 1:

            return e

e = find\_e(phi)

# Calculate d such that (d \* e) % phi = 1

d = mod\_inverse(e, phi)

print(f"RSA Model:")

print(f"Public key: ({e}, {n})")

print(f"Private key: ({d}, {n})")

# Diffie-Hellman Key Exchange

g = p  # Let's use the bigger prime as the base

alice\_private = random.randint(5, 10)  # A random private key for Alice

bob\_private = random.randint(10, 20)  # A random private key for Bob

alice\_public = g \*\* alice\_private

bob\_public = g \*\* bob\_private

alice\_secret = bob\_public \*\* alice\_private

bob\_secret = alice\_public \*\* bob\_private

print("\nDiffie-Hellman Key Exchange:")

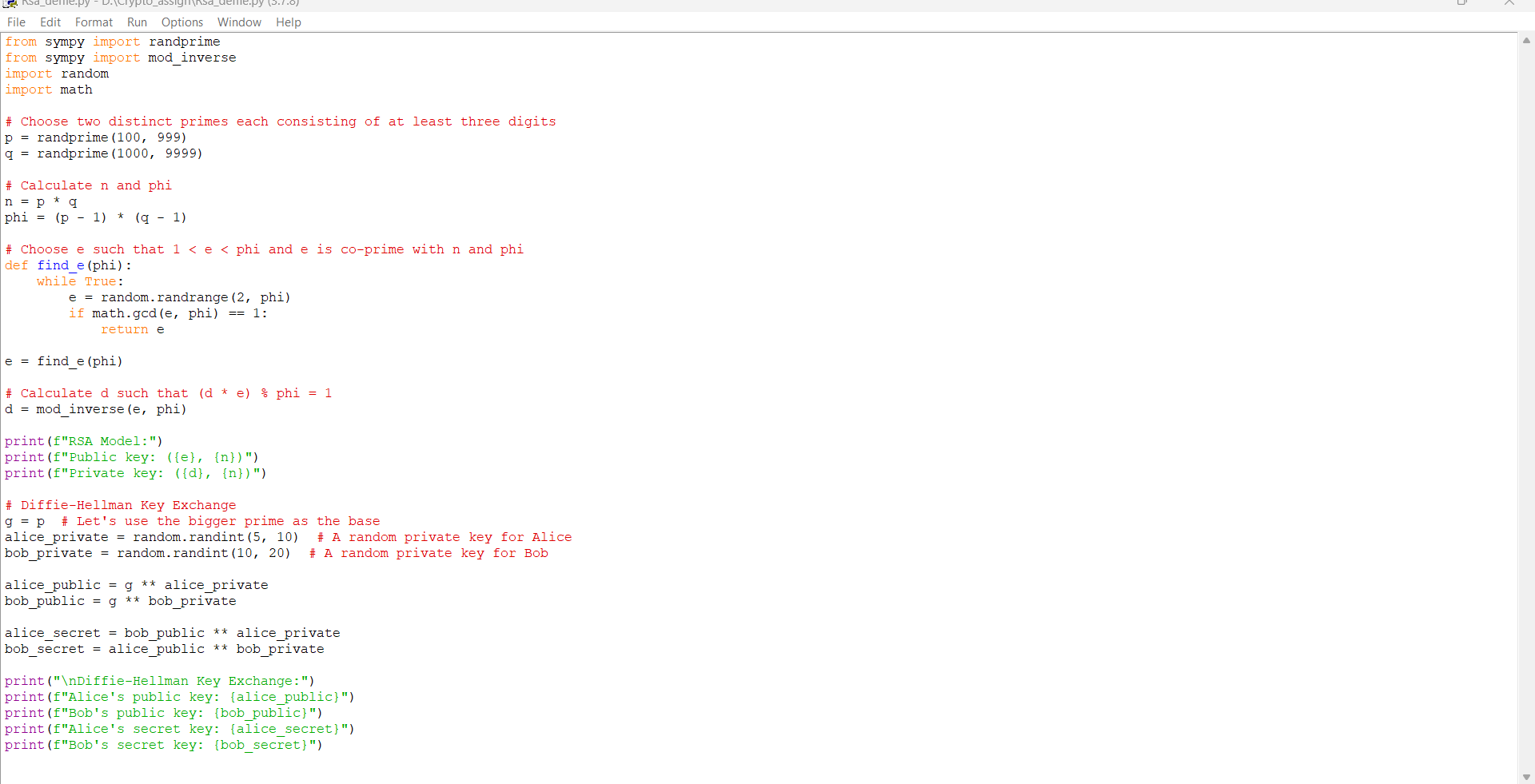
print(f"Alice's public key: {alice\_public}")

print(f"Bob's public key: {bob\_public}")

print(f"Alice's secret key: {alice\_secret}")

print(f"Bob's secret key: {bob\_secret}")

Code ScreenShots:-



Output:-

