

25. Write a python program for set of blocks encoded with the RSA algorithm and we don't have the private key. Assume $n = pq$, e is the public key. Suppose also someone tells us they know one of the plaintext blocks has a common factor with n . Does this help us in any way?

Code:

```
import math

# Suppose RSA modulus (unknown factors)

n = 3599 # = 59 * 61

e = 31

# Attacker receives ciphertext blocks, but also learns that

# one plaintext block has a common factor with n.

# Let's simulate such a plaintext:

m_bad = 59 # shares a factor with n

# Attacker computes gcd(m_bad, n)

g = math.gcd(m_bad, n)

print("gcd =", g)

if 1 < g < n:

    print("Non-trivial factor of n found:", g)

    p = g

    q = n // g

    print("p =", p)

    print("q =", q)

# Compute phi(n)

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

print("phi(n) =", phi)

# Compute private key using modular inverse

def egcd(a, b):

    if b == 0:

        return (a, 1, 0)

    g, x1, y1 = egcd(b, a % b)
```

```

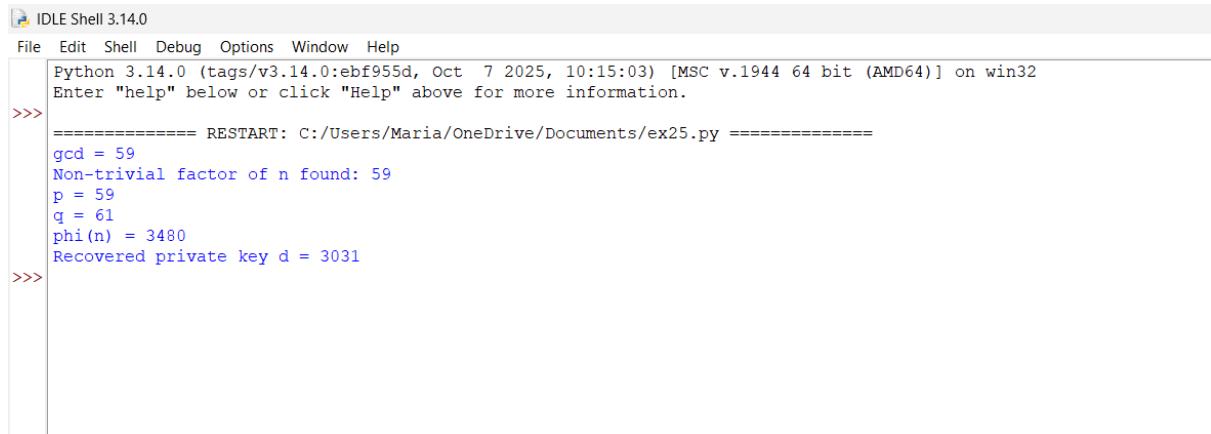
        return (g, y1, x1 - (a // b) * y1)

def modinv(a, m):
    g, x, y = egcd(a, m)
    if g != 1:
        raise ValueError("Inverse does not exist")
    return x % m

d = modinv(e, phi)
print("Recovered private key d =", d)

else:
    print("No help; gcd=1, plaintext is normal.")

```



```

IDLE Shell 3.14.0
File Edit Shell Debug Options Window Help
Python 3.14.0 (tags/v3.14.0:ebf955d, Oct 7 2025, 10:15:03) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.

>>> ===== RESTART: C:/Users/Maria/OneDrive/Documents/ex25.py =====
gcd = 59
Non-trivial factor of n found: 59
p = 59
q = 61
phi(n) = 3480
Recovered private key d = 3031
>>>

```