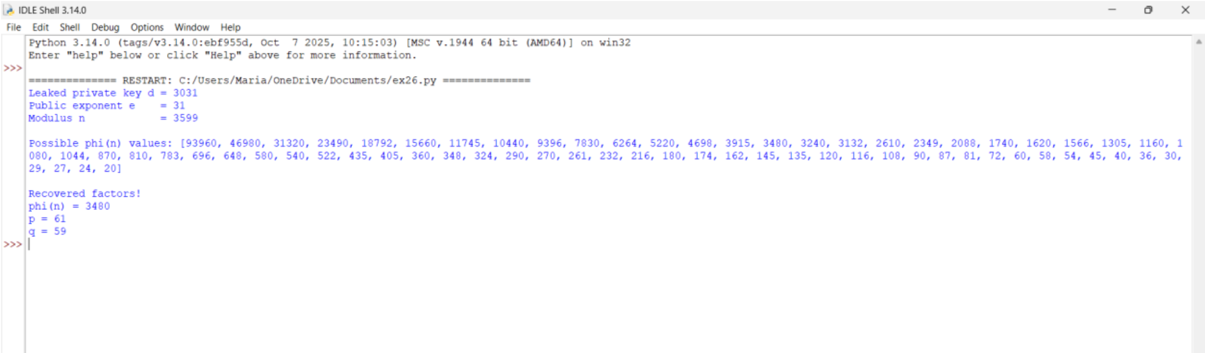
26. Write a python program for RSA public-key encryption scheme, each user has a public key, e, and a private key, d. Suppose Bob leaks his private key. Rather than generating a new modulus, he decides to generate a new public and a new private key. Is this safe?

Code:

```
import math

# Bob’s original RSA keys (public = e, private = d)

n = 3599       # Bob's modulus

e = 31         # public exponent

d = 3031       # Bob leaked this!

print("Leaked private key d =", d)

print("Public exponent e    =", e)

print("Modulus n            =", n)

# Attack: compute phi(n) from e and d

# ed = 1 (mod phi(n))  -> ed - 1 = k * phi(n)

ED_minus_1 = e*d - 1

phi_candidates = []

for k in range(1, 5000):

    if ED_minus_1 % k == 0:

        phi_candidates.append(ED_minus_1 // k)

print("\nPossible phi(n) values:", phi_candidates)

# For each phi, try to factor n using:

# p + q = n - phi(n) + 1

# pq = n

for phi in phi_candidates:

    S = n - phi + 1              # p + q

    D = S*S - 4*n                # discriminant

    if D >= 0:

        root = int(math.isqrt(D))

        if root * root == D:
```

```
p = (S + root) // 2

q = (S - root) // 2

if p*q == n:

    print("\nRecovered factors!")

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

    print("p =", p)

    print("q =", q)

    break
```



```
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/ex26.py ==============
Leaked private key d = 3031
Public exponent e    = 31
Modulus n            = 3599

Possible phi(n) values: [93960, 46980, 31320, 23490, 18792, 15660, 11745, 10440, 9396, 7830, 6264, 5220, 4698, 3915, 3480, 3240, 3132, 2610, 2349, 2088, 1740, 1620, 1566, 1305, 1160, 1
080, 1044, 870, 810, 783, 696, 648, 580, 540, 522, 435, 405, 360, 348, 324, 290, 270, 261, 232, 216, 180, 174, 162, 145, 135, 120, 116, 108, 90, 87, 81, 72, 60, 58, 54, 45, 40, 36, 30,
29, 27, 24, 20]

Recovered factors!
phi(n) = 3480
p = 61
q = 59
>>>
```