

**Write a Java/C/C++/Python program to implement RSA algorithm.**

**Program:**

```
# Python for RSA asymmetric cryptographic algorithm.
# For demonstration, values are
# relatively small compared to practical application
import math

def gcd(a, h):
    temp = 0
    while(1):
        temp = a % h
        if (temp == 0):
            return h
        a = h
        h = temp

p = int(input("Enter the Prime number P: "))
q = int(input("Enter the Prime number Q: "))
n = p*q
e = 2
phi = (p-1)*(q-1)

while (e < phi):

    # e must be co-prime to phi and
    # smaller than phi.
    if(gcd(e, phi) == 1):
        break
```

```

        else:
            e = e+1

# Private key (d stands for decrypt)
# choosing d such that it satisfies
#  $d \cdot e = 1 + k \cdot \text{totient}$ 

k = 2

d = (1 + (k*phi))/e

# Message to be encrypted
msg=int(input("Enter the Message which we wanted to encrypt : "))

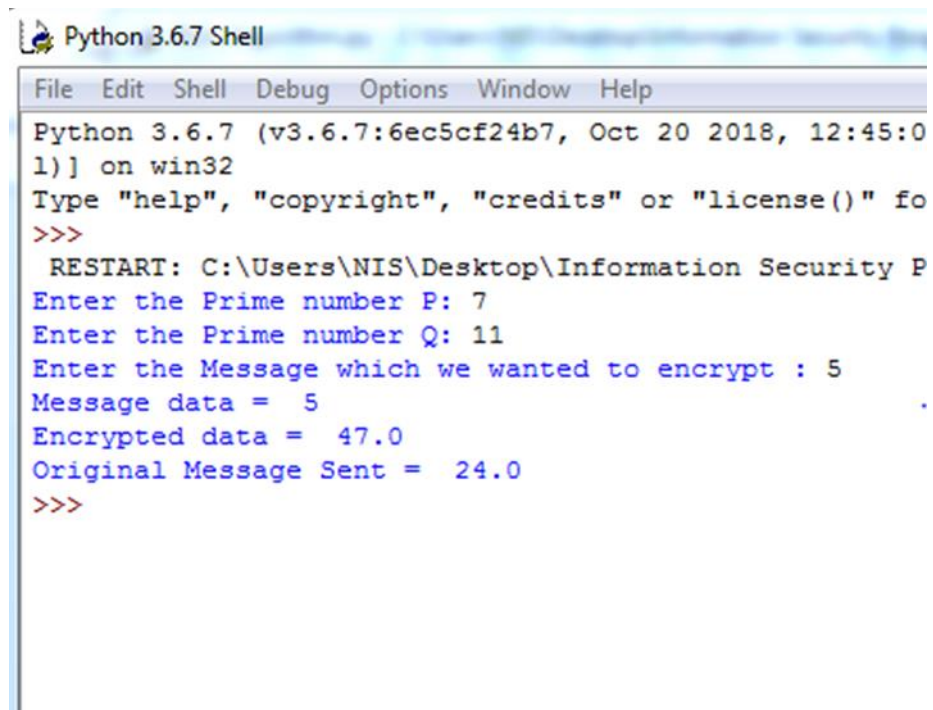
print("Message data = ", msg)

# Encryption  $c = (\text{msg}^e) \% n$ 
c = pow(msg, e)
c = math.fmod(c, n)
print("Encrypted data = ", c)

# Decryption  $m = (c^d) \% n$ 
m = pow(c, d)
m = math.fmod(m, n)
print("Original Message Sent = ", m)

```

## Output:



```
Python 3.6.7 Shell
File Edit Shell Debug Options Window Help
Python 3.6.7 (v3.6.7:6ec5cf24b7, Oct 20 2018, 12:45:01) on win32
Type "help", "copyright", "credits" or "license()" for more
>>>
RESTART: C:\Users\NIS\Desktop\Information Security P
Enter the Prime number P: 7
Enter the Prime number Q: 11
Enter the Message which we wanted to encrypt : 5
Message data = 5
Encrypted data = 47.0
Original Message Sent = 24.0
>>>
```