In [7]:

```
'''
Practical No. : 05
Program Code with output
Title : Write a Java/C/C++/Python program to implement RSA algorithm.

'''

def gcd(a,b):
    if(a==0):
        return b
    if(b==0):
        return a
    if(a==b):
        return a
    return gcd(b,a%b)

def isPrime(x):
    j = 2
    limit = x ** 0.5

    while (j <= limit):
        if (x % j == 0):
            return False
        j += 1

    return True


def accept_p_q():
    while (True):
        print("\nEnter two different Prime numbers : \n")

        while (True):
            p = int(input("Enter prime number p : "))

            if (isPrime(p)):
                break

        while (True):
            q = int(input("Enter prime number q : "))

            if (isPrime(q)):
                break

        if (p != q):
            break

    return p, q




def main():
    p, q = accept_p_q()

    n = p*q
```

```
    phi_n = (p-1) * (q-1)

    while(True):
        e = int(input("Select e such that e and phi_n are coprime and 1<e<phi_

        if(1<e and e<phi_n and gcd(e,phi_n)==1):
            break

    k=0

    while(True):
        d = (1+(k*phi_n)) / e

        if(int(d)==d):
            d = int(d)
            break

        k+=1



    print(f'Private key : ({d},{n})')
    print(f'Public key  : ({e},{n})')

    plain_text = int(input("Enter plain text : "))

    print("\nEncryption : ")
    print(f'Plain Text : {plain_text}')
    cipher_text = (plain_text**e)%n
    # cipher_text = pow(plain_text,e,n)
    print(f'Cipher Text : {cipher_text}')

    print("\nDecryption : ")
    print(f'Cipher Text : {cipher_text}')
    plain_text = (cipher_text**d)%n
    print(f'Plain Text : {plain_text}')

main()
```

```
Enter two different Prime numbers :

Enter prime number p : 53
Enter prime number q : 59
Select e such that e and phi_n are coprime and 1<e<phi_n : 3
Private key : (2011,3127)
Public key  : (3,3127)
Enter plain text : 89

Encryption :
Plain Text : 89
Cipher Text : 1394

Decryption :
Cipher Text : 1394
Plain Text : 89
```