**Cryptography & Network Security**

**PRN - 2019BTECS00026**

**Name - Niraja Vasudev Kulkarni**

**Batch - B1**

**Assignment - 8**

**Title**: Euclidean and Extended Euclidean Algorithm

**Aim:** To Demonstrate Euclidean and Extended Euclidean Algorithm

**Theory:**

In mathematics, the Euclidean algorithm, or Euclid's algorithm, is an efficient method for computing the greatest common divisor (GCD) of two integers (numbers), the largest number that divides them both without a remainder.

The extended Euclidean algorithm is particularly useful when a and b are coprime. With that provision, x is the modular multiplicative inverse of a modulo b, and y is the modular multiplicative inverse of b modulo a.

Code:

def Extended(a,b):

    r1=a

    r2=b

    t1=0

    Wt2=1

    while(r2>0):

        q = r1 // r2

        r = r1 % r2

        t = t1 - q \* t2

        r1 = r2

        r2 = r

        t1 = t2

        t2 = t

    if(t1<0):

        return t1+a

    return t1

a= int(input("Enter number M: "))

b= int(input("Enter number A: "))

inverse = Extended(a,b)

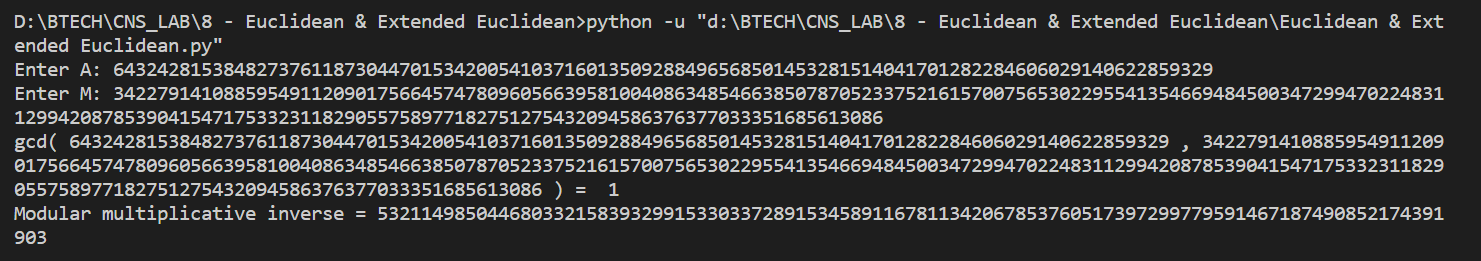
print("Multiplicative modular inverse - %d" %(inverse))

# A = 6432428153848273761187304470153420054103716013509288496568501453281514041701282284606029140622859329

# X = 5321149850446803321583932991533033728915345891167811342067853760517397299779591467187490852174391903

# M = 34227914108859549112090175664574780960566395810040863485466385078705233752161570075653022955413546694845003472994702248311299420878539041547175332311829055758977182751275432094586376377033351685613086

Output:



**Conclusion:**

The Euclidean and Extended Euclidean algorithm are used to find the GCD of numbers and the Multiplicative inverse of two coprime numbers respectively.