NAME: PRERNA SUNIL JADHAV
SAP ID: 60004220127
BATCH: C2-2
BRANCH: COMPUTER ENGINEERING
COURSE: INFORMATION SECURITY LABORATORY
COURSE CODE: DJ19CEL603

EXPERIMENT 05

AIM: Study and Implement RSA algorithm.

THEORY: RSA algorithm is an asymmetric cryptography algorithm. Asymmetric actually means that it works on two different keys i.e., Public key and Private key. As the name describes that the Public key is given to everyone and the Private key is kept private.

RSA process:

1) Choose 2 large prime no.s P and Q.
2) Calculate $N = P \times Q$
3) Select the public key (i.e., the encryption key) E such that it is not a factor of $(P-1)$ and $(Q-1)$
4) Select the decryption key (i.e., the private key) D such that the following equation is true:
   $(D \times E) \bmod (P-1) \times (Q-1) = 1$

5) For encryption, calculate the cipher text CT from the plain text PT as follows:

$CT = PT^E \bmod N$

6) send CT as cipher text to the receiver.

7) for Decryption, calculate the plain text PT

$PT = CT^D \bmod N$

Example:

1) let $P = 7$, $Q = 17$

2) $N = P \times Q$

$\therefore N = 7 \times 17 = 119$

3) select the public key E that it is not a factor of $(P-1)(Q-1)$

$(7-1)(17-1) = 6 \times 16 = 96$

lets choose E as 5 (as none of its factor is 3 and 2)

4) select the private key D.

$(D \times E) \bmod (P-1)(Q-1) = 1$

we have: $(D \times 5) \bmod (7-1)(17-1) = 1$

$(D \times 5) \bmod 96 = 1$

let $D = 77$, Then the following equation becomes true,

$(D \times 5) \bmod (96) = 1$

$\therefore (77 \times 5) \bmod (96) = 1$

$\therefore 385 \bmod 96 = 1$

5) Encyption: $CT = 10^5 \bmod 119 = \underline{40}$

6) Send 40 as ciphertext to receiver.

7) Decyption: $PT = 40^{77} \bmod 119 = \underline{10}$

CONCLUSION: Hence we studied and implemented RSA algorithm.

Name:	Prerna Sunil Jadhav
Sap Id:	60004220127
Class:	T. Y. B. Tech (Computer Engineering)
Course:	Information Security Laboratory
Course Code:	DJ19CEL603
Experiment No.:	05

AIM: Study and Implement RSA Algorithm.

CODE:

```
import math
def enc(plain,e,n):
    return (plain**e)%n
def dec(cipher,d,n):
    return (cipher**d)%n
def get_public_key(phi):
    e = 2
    while e < phi:
        if math.gcd(e,phi) == 1:
            break
        else:
            e += 1
    return e
def get_private_key(e,phi):
    d = 2
    while d < phi:
        if (d*e)%phi == 1:
            break
        else:
            d += 1
    return d
if __name__=='__main__':
    p,q = input('Enter two prime numbers: ').split()
    plain = int(input('Enter the plain text: '))
    p,q = int(p),int(q)
    n = p*q
    phi = (p-1)*(q-1)
    e = get_public_key(phi)
    d = get_private_key(e,phi)
    print('Public key(e,n): ',e,n)
    print('Private key(d,n): ',d,n)
    cipher = enc(plain,e,n)
    print('Cipher text: ',cipher)
    print('Plain text: ',dec(cipher,d,n))
```

OUTPUT:

```
PS C:\Users\Jadhav\Documents\BTech\Docs\6th Sem\IS\Code> & C:/msys64/mingw64/bin/python.exe "c:/Users/Jadhav/Doc
uments/BTech/Docs/6th Sem/IS/Code/Exp5/RSA.py"
Enter two prime numbers: 1291 607
Enter the plain text: 909
Public key(e,n):  7 783637
Private key(d,n):  670063 783637
Cipher text:  359730
Plain text:  909
PS C:\Users\Jadhav\Documents\BTech\Docs\6th Sem\IS\Code>
```