

RSA Algorithm

Cryptography – Lab 4

Name: Ojas Patil
Reg No: 21BAI1106

Task

To Develop a Python-based RSA Algorithm implementation.

RSA Algorithm - Definition

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 Algorithm - Output Snapshot



```
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ojasa>cd C:\VIT\sem 7\crypto lab\lab4

C:\VIT\sem 7\crypto lab\lab4>python rsa_21BAI1106.py
n = 21
e = 5
d = 5.0
Public key: (5, 21)
Private key: (5.0, 21)
Original message:11
Encrypted message: 2.0
Decrypted message: 11.0

C:\VIT\sem 7\crypto lab\lab4>
```

RSA - Handwritten sum

classmate
Date _____
Page _____

RSA

$n = p \cdot q$ $e \cdot d$

$$\begin{aligned} C &= P^e \bmod n \\ P &= C^d \bmod n \end{aligned}$$

Q) Perform encryption of $msg = 8$ using RSA
 $p = 7, q = 11, n = 77, e = 17$

Solⁿ

$n = p \cdot q$
 $n = 7 \cdot 11$
 $n = 77$

$\phi = (p-1)(q-1)$
 $= 6 \cdot 10$
 $\phi = 60$

Step 1

$d \cdot e \bmod \phi = 1$
 $d \cdot 17 \bmod 60 = 1$

$d = 11$

$d = 55$

Step 2

$C = P^e \bmod n$
 $= 8^{17} \bmod 77$
 $= 8^{16} \cdot 8 \bmod 77 = 36 \cdot 8 \bmod 77$

$C = 57$

← cipher text

$8^1 \bmod 77 = 8$
 $8^2 \bmod 77 = 64$
 $8^3 \bmod 77 = 64 \cdot 8 \bmod 77 = 15$
 $8^4 \bmod 77 = 15 \cdot 8 \bmod 77 = 71$
 $8^5 \bmod 77 = 71 \cdot 8 \bmod 77 = 36$

Source Code

```
import math

# step 1
p = 3
q = 7

# step 2
n = p*q
print("n =", n)

# step 3
phi = (p-1)*(q-1)

# step 4
```

```
e = 2
while(e<phi):
    if (math.gcd(e, phi) == 1):
        break
    else:
        e += 1

print("e =", e)
# step 5
k = 2
d = ((k*phi)+1)/e
print("d =", d)
print(f'Public key: {e, n}')
print(f'Private key: {d, n}')

# plain text
msg = 11
print(f'Original message:{msg}')

# encryption
C = pow(msg, e)
C = math.fmod(C, n)
print(f'Encrypted message: {C}')

# decryption
M = pow(C, d)
M = math.fmod(M, n)

print(f'Decrypted message: {M}')
```

Conclusion

The implementation of RSA Algorithm in Python successfully demonstrates the core components of the algorithm and provides a foundational understanding of RSA and its practical application in securing data.