Kartik Padave

A022

70362019039

Experiment 6

Aim

To implement RSA.

Theory

RSA algorithm is asymmetric cryptography algorithm. Asymmetric 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 Private key is kept private. The idea of RSA is since it is difficult to factorize a large integer. The public key consists of two numbers where one number is multiplication of two large prime numbers. And private key is also derived from the same two prime numbers. So, if somebody can factorize the large number, the private key is compromised. Therefore, encryption strength totally lies on the key size and if we double or triple the key size, the strength of encryption increases exponentially. RSA keys can be typically 1024 or 2048 bits long, but experts believe that 1024-bit keys could be broken in the near future. But till now it seems to be an infeasible task.

Code

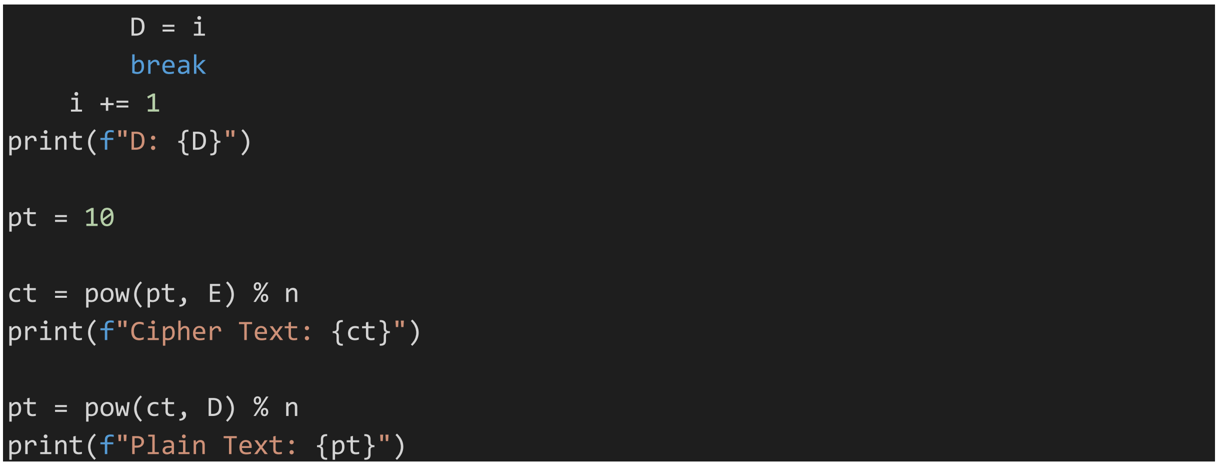
```
p = int(input("Enter prime no. (p): "))
q = int(input("Enter prime no. (q): "))

n = p*q
print(f"n: {n}")

tot = (p-1)*(q-1)
print(f"Totient of n: {tot}")

print("Note: Value of E is 1 < E < totient of n & is not a factor of totient of n")
E = int(input("Enter (E): "))

i = 1
while True:
    if(i*E % tot == 1):
```



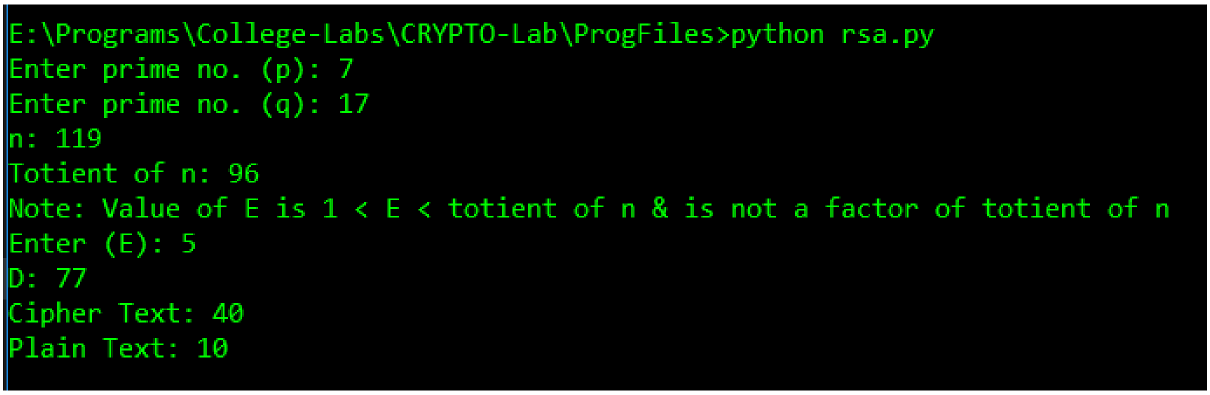
```
        D = i
        break
    i += 1
print(f"D: {D}")

pt = 10

ct = pow(pt, E) % n
print(f"Cipher Text: {ct}")

pt = pow(ct, D) % n
print(f"Plain Text: {pt}")
```

Output



```
E:\Programs\College-Labs\CRYPTO-Lab\ProgFiles>python rsa.py
Enter prime no. (p): 7
Enter prime no. (q): 17
n: 119
Totient of n: 96
Note: Value of E is 1 < E < totient of n & is not a factor of totient of n
Enter (E): 5
D: 77
Cipher Text: 40
Plain Text: 10
```

Conclusion

Hence, we were able to implement RSA.