CS152-Homework8

Hongchen Cao 2019533114

2020.11.16

1

1.1

$$
\begin{aligned}
x &\equiv M_p q x_p + M_q p x_q \ (mod\ n) \\
&\equiv M_p q x_p \ (mod\ p) \\
&\equiv q^{-1} q y^{d_p} \ (mod\ p) \\
&\equiv y^{d_p} \ (mod\ p) \\
&Because\ d_p \equiv d \ (mod\ p-1) \\
&\equiv y^{d} \ (mod\ p)
\end{aligned}
$$

Similarly we have $x \equiv y^d \ (mod\ q)$

Thus, we have $x \equiv y^d \ (mod\ n)$

1.2

d_p: 907

d_q: 1345

M_p: 777

M_q: 973

```
from sage.all import *

p = 1511
q = 2003
d = 1234577
dp = pow(d, 1, p - 1)
dq = pow(d, 1, q - 1)
Mp = inverse_mod(q, p)
Mq = inverse_mod(p, q)

print("dp:", dp)
print("dq: ", dq)
print("Mp: ", Mp)
print("Mq: ", Mq)
```

1.3

x: 1443247

```
y = 152702
xp = pow(y, dp, p)
xq = pow(y, dq, q)
x = pow(Mp * q * xp + Mq * p * xq, 1, p * q)
print("x:", x)
```

2

$(\frac{610}{987}) = 1$
$(\frac{20964}{1987}) = -1$
$(\frac{1234567}{11111111}) = 1$

```
def jacobi(a: int, n: int) -> int:
    a %= n
    res = 1
    while a != 0:
        while a % 2 == 0:
            a /= 2
            n_mod_8 = n % 8
            if n_mod_8 in (3, 5):
                res = -res
        a, n = n, a
        if a % 4 == 3 and n % 4 == 3:
            res = -res
        a %= n
    if n == 1:
        return res
    else:
        return 0


print(jacobi(610, 987))
print(jacobi(20964, 1987))
print(jacobi(1234567, 11111111))
```

3

B_1 =13

B_2 =47

```
from sympy.ntheory import pollard_pm1, primefactors

for i in range(3, 20):
    if pollard_pm1(262063, B=i) is not None:
        print(i)
        break
for i in range(3, 50):
    if pollard_pm1(9420457, B=i) is not None:
        print(i)
        break
```

4

Not prime

```
import random


def MillerRabin(n: int) -> bool:
    k = 0
    m = 0
    while True:
        m = (n - 1) // pow(2, k)
        if pow(2, k) * m == n - 1 and pow(int(m), 1, 2) == 1:
            break
        k += 1

    b = pow(random.randint(1, n), int(m), n)
    if b == 1:
        return True
    for i in range(0, k):
        if b == n - 1:
            return True
        else:
            b = pow(b, 2, n)
    return False


n = '1739788978351496189648308695359876353235148922052809145 73' \
    '2115619724269679679085240375597891487417120617083784812 0' \
    '5017865753096554838058038306206137432977766565647608464326' \
    '8296616275032787115289681130280929316788652639721116664 9' \
    '8370475757652374822169740788077289029083381001639360536 8169' \
    '4945358390189163179112628719562422887803517661727450414' \
    '6394448620505476647251926632748223972512934907559647678 9814' \
    '0661528504879782380364478623121817524321484051307847497' \
    '2799108333261435785431915660660121001469470564194597979 0601' \
    '8918465410663341843118614746568200081683099400746255691' \
    '9624463511739770769413026977709450003624902245 29'
n = int(n)
for i in range(1, 500):
    if MillerRabin(n):
        print(True)
print(False)
```