Assignment #4

Matthew Langlois - 7731813

Dec. 4

Question 1

1. Using sage math I was able to code a solution to solve RSA for larger numbers:

```
# Message as per the assigment
m = 399621883454709

# Given public exponent
e = 65537

# Prime number generation (a number between 2^1024 and 2^1025 which is
 ↪  prime)
p = random_prime(2^1025,None,2^1024)
q = random_prime(2^1025,None,2^1024)

# Compute phi and n
n = p*q
phi = (p-1)*(q-1)

# Compute the private exponenet
d = inverse_mod(e,phi)

# Ecnrypt the data
encrypted = power_mod(m,e,n)

# Decrypt the data normally
timeit('pow(encrypted,d,n)')
print(pow(encrypted,d,n))


# Decrypt the data using chinese remainder therom
dP = int(d)%(p-1)
dQ = int(d)%(q-1)

cP = c%p
cQ = c%q
```

```
mP = power_mod(cP, dP, p)
mQ = power_mod(cQ, dQ, p)

timeit('crt([mP, mQ], [p,q])')
print(crt([mP, mQ], [p,q]))
```

When decrypting without Chinese remainder theorem we get: 125 loops, best of 3: 6.72 ms per loop.

When decrypting using the Chinese remainder theorem we get: 125 loops, best of 3: 7.43 ms per loop.

Question 2

Let $p = 1019$

Let $q = 1231$

Let $M = p \cdot q = 1019 \cdot 1231 = 1254389$
Let $s = 1254383$ since 1254383 and 1254389 are co-prime.
Using the BBS PRNG algorithm $x_{n+1} = x_n^2 \mod m$ we get the following numbers

x_0	x_1	x_2	x_3	x_4
36	1296	425227	335957	946796
x_5	x_6	x_7	x_8	x_9
1163324	68546	867311	138368	1218506
x_{10}	x_{11}	x_{12}	x_{13}	x_{14}
586575	108648	587414	390054	1044273
x_{15}	x_{16}	x_{17}	x_{18}	x_{19}
512601	412593	1106848	934364	156331

Note that this table was generated using the following code:

```
let p = 1019;
let q = 1231;
let M = p*q;
let seed = 1254383;

for(let i = 0; i < 20; i++) {
    seed = (seed*seed)%M;
    console.log(seed)
}
```

For real world security it would be ideal to use primes of size 1024 this way the modulus is of size 2048 bits. This would provide the currently minimum recommended security level for

confidentiality. To be more secure it would be better to use 2048 bit primes this way the modulus is 4096 bits in length.

To have equivalent security to AES-128 the bit length of the modulus would need to be 3072 bits. Thus p and q will each need to be 1536 bits in length ($3072/2 = 1536$).

Question 3

Using the generator function we get the following bit sequences:

$g^0 = 00001$
$g^1 = 00010$
$g^2 = 00100$
$g^3 = 01000$
$g^4 = 10000$
$g^5 = g^2 + 1 = 00101$
$g^6 = g^5 \cdot g^1 = (g^2 + 1) \cdot g^1 = g^3 + g^1 = 01010$
$g^7 = g^6 \cdot g^1 = (g^3 + g^1) \cdot g^1 = g^4 + g^2 = 10100$
$g^8 = g^7 \cdot g^1 = (g^4 + g^2) \cdot g^1 = g^5 + g^3 = 01101$
$g^9 = g^8 \cdot g^1 = (g^5 + g^3) \cdot g^1 = g^6 + g^4 = 11010$
$g^{10} = g^9 \cdot g^1 = (g^6 + g^4) \cdot g^1 = g^7 + g^5 = 10001$
$g^{11} = g^{10} \cdot g^1 = (g^7 + g^5) \cdot g^1 = g^8 + g^6 = 00111$
$g^{12} = g^{11} \cdot g^1 = (g^8 + g^6) \cdot g^1 = g^9 + g^7 = 01110$
$g^{13} = g^{12} \cdot g^1 = (g^9 + g^7) \cdot g^1 = g^{10} + g^8 = 11100$
$g^{14} = g^{13} \cdot g^1 = (g^{10} + g^8) \cdot g^1 = g^{11} + g^9 = 11101$
$g^{15} = g^{14} \cdot g^1 = (g^{11} + g^9) \cdot g^1 = g^{12} + g^{10} = 11111$
$g^{16} = g^{15} \cdot g^1 = (g^{12} + g^{10}) \cdot g^1 = g^{13} + g^{11} = 11011$
$g^{17} = g^{16} \cdot g^1 = (g^{13} + g^{11}) \cdot g^1 = g^{14} + g^{12} = 10011$
$g^{18} = g^{17} \cdot g^1 = (g^{14} + g^{12}) \cdot g^1 = g^{15} + g^{13} = 00011$
$g^{19} = g^{18} \cdot g^1 = (g^{15} + g^{13}) \cdot g^1 = g^{16} + g^{14} = 00110$
$g^{20} = g^{19} \cdot g^1 = (g^{16} + g^{14}) \cdot g^1 = g^{17} + g^{15} = 01100$
$g^{21} = g^{20} \cdot g^1 = (g^{17} + g^{15}) \cdot g^1 = g^{18} + g^{16} = 11000$
$g^{22} = g^{21} \cdot g^1 = (g^{18} + g^{16}) \cdot g^1 = g^{19} + g^{17} = 10101$
$g^{23} = g^{22} \cdot g^1 = (g^{19} + g^{17}) \cdot g^1 = g^{20} + g^{18} = 01111$
$g^{24} = g^{23} \cdot g^1 = (g^{20} + g^{18}) \cdot g^1 = g^{21} + g^{19} = 11110$
$g^{25} = g^{24} \cdot g^1 = (g^{21} + g^{19}) \cdot g^1 = g^{22} + g^{20} = 11001$
$g^{26} = g^{25} \cdot g^1 = (g^{22} + g^{20}) \cdot g^1 = g^{23} + g^{21} = 10111$
$g^{27} = g^{26} \cdot g^1 = (g^{23} + g^{21}) \cdot g^1 = g^{24} + g^{22} = 01011$
$g^{28} = g^{27} \cdot g^1 = (g^{24} + g^{22}) \cdot g^1 = g^{25} + g^{23} = 10110$

$g^{29} = g^{28} \cdot g^1 = (g^{25} + g^{23}) \cdot g^1 = g^{26} + g^{24} = 01001$
$g^{30} = g^{29} \cdot g^1 = (g^{26} + g^{24}) \cdot g^1 = g^{27} + g^{25} = 10010$
$g^{31} = g^{30} \cdot g^1 = (g^{27} + g^{25}) \cdot g^1 = g^{28} + g^{26} = 00001$

The point at infinity is: 00000.