Mehmet Yağız Çebişli – 28229

CS411 -HW2

1-

1. n -> 394 , t -> 49 The number of elements in the group is the count of the numbers

that are relatively prime with 394 -> phi(394)

1. In order to be a generator a number must be relatively prime with n, let's say that number is 3
2. A subgroup n will contain elements in form of 49\*k + n (n < 49) so all of them are relatively prime with 49 so the answer is 49.

2-

d = inv(e) mod phi(n) and m = c^d mod n 🡪 m = c^inv(e) mod n

3- In ciphertext 1 none of the bytes are corrupted and the message is “In the Imaginary world of Witchcraft and Wizardry, You pick side either with Grindelwald or Dumbledore”

In ciphertext 2 3 bytes of nonce were corrupted. The decrypted message is: “This might be a wrong decryption of the message. It is up to you to\ J”

In ciphertext 3 7 bytes of nonce were corrupted. The decrypted message is: “Do what is right not what is easy”

4- The idea is first I checked the gcd of a and n. If that is 1 there is only one solution which is inv(a) \* b mod n. Otherwise, we need to check whether b is divisible by gcd(a,b) or not. If gcd divides b there are gcd solutions. Otherwise, no solutions. Python implementation is also available in the solution.py file.

a)

There is only one solution and that is: 778214478105812676636756719791275483076257993476674633986349

b)

There is no solution.

c)

There are 2 solutions, and they are: 214452804613963486637685100244716357445929679678527437334799, 510140495723613606819499501310772970562507011340109785493345,

d)

There are 4 solutions, and they are: 896049914695400631894506737412603426006253201283161992061, 18951860339961353869399140376877946054778848148808512468685, 37007670765227307106903774016343288683551443096333862945309, 55063481190493260344408407655808631312324038043859213421933,

5- LFSR.PY functions have been used.

Maximum period sequence achieved for x6 + x5 + x4 + x + 1: 63

First period for x6 + x2 + 1: 14 which is smaller than 63 so it is not maximum

First period for x5 + x3 + 1: 1 which is smaller than 31 so it is not maximum

6- LFSR.PY functions have been used.

X1 -> L and C(x): (45, [1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1]) len x1 89

X2 -> L and C(x): (29, [1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]) len x2 87

X3 -> L and C(x): (37, [1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]) len x3 102

#Expected linear complexity of a random sequence E(L(sn)) ≈ n/2 + 2/9.

# For x1 L = 45 and n/2 + 2/9 ~= 44.7 so it can be considered as random.

# For x2 L = 29 and n/2 + 2/9 ~= 43.7 so it can be considered as predictable

# For x3 L = 37 and n/2 + 2/9 ~= 51.2 so it can be considered as predictable.

7- We can not decrypt the message because it is impossible to go backward in the message. If we knew the beginning of the plaintext we could decrypt the message.