**Theory Homework Assignment 2**

Student Name: - Tolulope Olugbenga

Student Number: - 3643581

**Questions/ Answers**

1a.) is a prime number, if  then prove that?

By Fermat , then  in modulo p

 = 

In modulo 

Since  and  are divisible by. The product is divisible by.

1b.) Let, then prove that?

Since the  by Euler’s theorem  and. But  and. Therefore,  and  .

Therefore, 

2.) a. Boss A and his set of friends B = {b1,b2,…, bn} collectively share a symmetric key EAES-ABi

b. A encrypts each member of the set of keywords, K = {k1, k2, …, kn}, with EAES-ABi to obtain K’ = {E(k1), E(k2),…, E(kn)}.

c. A encrypts each member of the set of friends, B = {b1,b2,…, bn}, with EAES-ABi to obtain B’ = {E(b1), E(b2),…, E(bn)}.

d. To create the Bloom filter for K’, A creates an array L of size q and r hash functions {h1, h2,…, hr | hi : K’ ->{0,1,…, q-1}}, such that A initially sets L to 0, for any E(k) ∈ K’, and then A sets L[hi(E(k))] = 1 for 1 ≤ i ≤ r.

e. To create the Bloom filter for B’, A creates an array R of size q and s hash functions {h1, h2,…, hs | hi : B’ ->{0,1,…, q-1}}, such that A initially sets R to 0, for any E(b) ∈ B’, and then A sets R[hi(E(b))] = 1 for 1 ≤ i ≤ s .

f. A sends L, q hash functions, R and s hash functions to Secretary S.

g. A friend bi forwards a message, with set of encrypted keywords P = {E(k1) …, E(km)} and his/her encrypted information, E(bx), sent to S.

h. S will check to see if E(bx) ∈ B’ by checking whether all locations of R[hi(E(bx))], for 1 ≤ i ≤ s, is set to 1. If this is true, S will check for the keyword or will discard the message if otherwise.

i. S will check if E(kj) ∈ K’, for 0 ≤ j ≤ m, by checking whether all locations of L[hi(E(kj))], for 1 ≤ i ≤ r, is set to 1. If this is true, S will forward the message or will discard the message if otherwise.

j. A and S can also share a symmetric key, which is different from EAES-ABi, to encrypt communication if necessary.

3.) a. For each element in the set A = {a1, a2, … an}, Alice computes two random numbers per bit of the element to obtain a set R = {{α0, α1, β0, β1,…}1, {α0, α1, β0, β1,…}2, …, {α0, α1, β0, β1,…}n}.

b. For each element ai and based on the values of each bit in ai Alice selects the appropriate random numbers from R, and XORs them to generate a set of values A’ = {x1, x2, x3…xn} where α{0,1} ⊕ β{0,1}… = xi for all for 1 ≤ i ≤ n.

c. Alice sends A’ and R to Bob and Carlos.

d. For each element bi, in the set B = {b1, b2,, … bn } and based on the values of each bit in bi, Bob selects the appropriate random numbers from R, and XORs them to generate a set of values B’ = {y1, y2, y3…yn} where α{0,1} ⊕ β{0,1}… = yi for all for 1 ≤ i ≤ n.

e. Bob compares B’ and A’ to find similar values to generate A∩B

f. Bob sends A∩B to C

g. For each element, ci, in the set C = {c1, c2,, … cn } and based on the values of each bit in ci, Carlos selects the appropriate random numbers from R, and XORs them to generate a set of values C’ = {z1, z2, z3…zn} where α{0,1} ⊕ β{0,1}… = zi for all for 1 ≤ i ≤ n.

h. Carlos compares C’ to A∩B to find similar values to generate A∩B∩C.

4.)  a. Assuming Alice has input *x = 001* and Bob has input *y = 011*

1. Bob prepares to two random numbers per bit of *y = 011*. For *0*, *α0* and *α1* are created. For *1*, *β0* and *β1* are created. For *1*, *γ0* and *γ1* are created.
2. Bob sends α0, α1,β0,β1, γ0 and γ1 to Alice.
3. Based on the value each bit Bob has in *y = 011*, Bob selects the corresponding random number from each pair. For *0*, *α0* is selected; for *1*, *β1* is selected; and for *1*, *γ1* is selected.
4. Bob computes *C = α0* *⊕ β1* *⊕ γ1* and sends *C* to Alice.
5. Based on the value each bit Alice has in *x = 001*, Alice selects the corresponding random number from each pair. For *0*, *α0* is selected; for *0*, *β0* is selected; and for *1*, *γ1* is selected.
6. Alice computes *D = α0 ⊕ β0 ⊕ γ1*
7. Alice compare *C*and *D* to determine whether *C = D.*If they are equal, comparison is completed.
8. If they are not equal, both Alice and Bob divide x and y into two parts.
9. Steps b to h are repeated on the most significant section of the x and y.
10. If they are equal, steps b to h are repeated on the least significant section of the x and y.
11. For either step j or k, if the bits are not equal, steps i to j are repeated for either section until only two bits are compared.
12. If they are not equal, Alice and Bob will expose their bits, the person with a value of 1 has the highest value.
13. If they are equal, Alice and Bob will move bitwise towards the right until unequal result is obtained, for which step m will be performed to see who has the highest value.