CS458 Assignment 3

Name: Ron Ng Jian Ying

UW userid: r32ng

Student Number: 20801876

Question 1

1a)

The XOR of the two plaintexts is equivalent to the XOR of the two ciphertexts.

1b)

Crib dragging is the solution. Crib dragging can be used as the XOR of the two ciphertexts is the XOR of the two plaintexts.

For this question, I used a crib dragging program. What it does is to take a word (a crib), convert it to hex. Then, divide the XOR of the two plain texts (XORP) into equal segments of X bytes where X is the byte size of the crib. The program then XORs the crib over all the segments of the XORP and prints it in the console. If the crib of one plaintext is at the exact same position as one of the segments, then the program will display the other plaintext's decrypted output.

We need to do this several times, dragging out the inputs as required to fully decrypt the text.

Question 2

2a)

Bob's secret key is a = 23

Alice's secret key is b = 58

Alice gives to Bob the value A = g^a mod p = 5^{23} mod 83 = 19

Bob gives to Alice the value B = g^b mod p = 5^{58} mod 83 = 7

s = B^a mod p = 7^{23} mod 83 = **37**

$s = A^b \bmod p = 19^{58} \bmod 83 =$ **37**

Without divulging their secret keys, Alice and Bob were able to arrive at the same value s. S would be the secret key shared between the two parties.

2b)
Note that $B^a \bmod p$ is really $(g^b \bmod p)^a \bmod p = g^{ab} \bmod p$ and A^b is really $(g^a \bmod p)^b = g^{ab} \bmod p$.

Eve knows g, g^a and g^b, she needs to find the value of g^{ab} to obtain the secret key.
To find a, Eve would have to compute $\log_g(A) = b$. For certain groups of G, the discrete log can be quickly computed but if G is chosen properly it would be extremely difficult as there is no efficient way of solving for b.

Hence, Eve can recover the original secret values provided that G is not properly chosen. It is difficult to recover if G is chosen properly.

2c)
Alice ---- Mallory ---- Bob

Alice sends $g^a \bmod p$. Mallory intercepts and replaces g^a with g^m.
Bob sends $g^b \bmod p$. Mallory intercepts and replaces g^b with g^m.

Neither Alice or Bob knows that the values they received were altered by Mallory.

Mallory then computes $(g^a \bmod p)^b \bmod p$ which is the secret key with Alice and $(g^b \bmod p)^a$ which is the secret key with Bob.

When Alice sends messages to Bob, Mallory is able to decrypt the message since Alice shares a secret key with Mallory and not with Bob directly. The message can be altered and sent to Bob with the secret key shared by Mallory and Bob.

From Alice and Bob's perspective, nothing appears fishy.

A simple way to prevent this is to let g^a and g^b be signed by a Certificate Authority so that its values can be checked upon receiving it to make sure it is authentic.

Question 3

3a)
$n = 4g^2ab + 2ga + 2gb + 1 = 2g\ (2gab + a + b) + 1$

3b)

3c)

Question 4

a)
Assumption
Values of one column must be at least binary in nature like gender so that we can select two different groups.

Tracker
T = SELECT SUM (Salary) From Employee WHERE Gender = “F”

Query

q(C or T) = SELECT SUM (Salary) From Employee WHERE Name = “Cori” OR Gender = “F”

q(C or not T)
SELECT SUM (Salary) From Employee WHERE Name = “Cori” OR Gender <> “F”

q(S) = SELECT SUM (Salary) From Employee

Cori salary = q(C or T) + q(C or not T) – q(S)

4b)

TARGET is the salary that we are guessing Rachel will have.

q(C or T) = SELECT COUNT(*) FROM Employee WHERE Name = “Rachel” AND Salary = TARGET OR Gender = “F”

q(C or not T) = SELECT COUNT(*) FROM Employee WHERE Name = “Rachel” AND Salary = TARGET OR Gender <> “F”

q(S) = SELECT COUNT(*) FROM Employee WHERE Salary >= 0

q (C) = q(C or T) + q (C or not T) – q(S)

If q(C) = 2 i.e. COUNT returns 2, that means our guess was correct.