Cryptography - **Partner A**

Encrypting a 3 letter word with a One Time Pad:

- do the first two questions with your partner

1) Agree on a secret key with your partner (fill this in for k)
2) If encryption is done via XOR, how should decryption be done? (fill this in in the blank after "Decryption")

- do the next question by yourself

3) Choose a **secret** message that is 3 letters long and fill in the binary below.

Encryption: XOR

k																								
m																								

Cipher text:

c																								

4) Tell your partner the resulting cipher text

- next, you will repeat the exercise but your partner will encode a message

5) Agree on a different secret key with your partner (fill this in for k)
6) Wait for the cipher text from your partner.
7) Decode the cipher text to recover the original message.

Decryption: ________ (what is the inverse of XOR?)

k																								
c																								

Message:

m																								

What is the message (in ascii)?

One Time Pad security + usage

1) What happens if we use the same One Time Pad twice? What information would an eavesdropper have access to?

2) What information about the messages would the eavesdropper be able to recover? (What is *c1* XOR *c2* equivalent to? Where *c1* is the first cipher text and *c2* is the second cipher text)

3) If I have a message of n bits, how many bits must my One Time Pad be?

Public/Private Keys (Diffie-Hellman exchange)

1) pick a prime number, *p.*

2) pick a primitive root modulo *p*. This is an integer *r* between [1, *p* - 1] such that the values of (*r* ^ *x*) % *p* for all *x* in range [0, *p* - 2] are different.

3) Choose a secret number. Write it down. (this is your private key)

4) Compute your public key. Do this by computing *r* ^ (*private key*) % *p*. Tell your partner your public key. Write down the public keys here.

5) Compute your shared secret. Do this by computing (*your partner's public key*) ^ (*your private key*) % *p*.