Homework 1

1 Problem 1 (12 points)

(a) Show that a Homophonic substitution cipher that encrypts a single character at a time can *never* have perfect secrecy, even for two-character messages, no matter how large the output alphabet is. That is, provide two different two-character messages such that the distributions of encryptions of those two messages are different.

(b) Show that the Vigenère cipher does *not* have perfect secrecy if the message length is even one character longer than the key length.

(c) Show that the Vigenère cipher does *not* have perfect secrecy if the keyspace is grammatically correct English, even if the message length is smaller than the key length.

(d) Show that the following cipher has perfect secrecy. Messages are ℓ bit strings. The key is a random permutation P on 2ℓ items. To encrypt a message m, write down m, followed by $\overline{m}$, the bitwise complement of m. Then permute the bits of the resulting 2ℓ-bit string $m||\overline{m}$ according to the permutation described by the key.

2 Problem 2 (12 points)

(a) Devise an encryption scheme such that (1) given an encryption of any message, an adversary can figure out 90% of the secret key, but (2) the scheme is still perfectly secure, despite 90% of the key being revealed. Do not forget to prove that the scheme is secure and that it is correct.

(b) Devise an encryption scheme such that (1) given an encryption of any message, an adversary learns *nothing* about the secret key, but (2) the scheme is completely broken (as in, given the ciphertext, an adversary can completely recover the plaintext).

3 Problem 3 (10 points)

Suppose we say that two messages are *adjacent* if they differ by at most a single bit.

Definition 1. *An encryption scheme* $(\mathsf{Enc}, \mathsf{Dec})$ *for* ℓ*-bit messages has* adjacent-message perfect secrecy *if, for any two* ℓ*-bit* adjacent *messages* m_0, m_1*, the distributions* $\mathsf{Enc}(k, m_0)$ *and* $\mathsf{Enc}(k, m_1)$ *are identical.*

This is the same definition as perfect secrecy seen in class, except for the restriction that it only applies to m_0, m_1 that are adjacent. Therefore, it is a seemingly weaker definition.

Prove that any encryption scheme that has *adjacent-message perfect* secrecy must in fact have *perfect* secrecy.
Hint: suppose m_0, m_1 *differed in just 2 bits. How would you prove that the distributions of their encryptions are identical? Generalize this to arbitrary messages.*

4 Problem 4 (16 points)

Consider the following encryption scheme. The key will be a table such as

	0	1	2	3	4	5	6	7	8	9
	g	z	a		t	k	f	w		s
3	c	m	e	.	b	x	p	u	h	y
8	i	d	v	r	_	q	j	l	n	o

The plaintext alphabet will consist of the 26 letters, plus spaces (represented by _) and periods. In general, the table will consist of 4 rows. The numbers 0-9 are always written in the first row, in increasing order. We will call these the column indices. Then 8 plaintext characters are placed in the first row, leaving two black spaces. The numbers from the first row corresponding to those spaces (in our example, 3 and 8) are then written in the third and forth row of the first column, again in increasing order. These will be called the row indices. The remaining 20 characters fill out the rest of the third and fourth row.

Encryption is character by character. For each character, find the character in the table. There are two cases:

- If the character is in the second row (the first row of non-numbers), that character is encrypted by its column index. So, for example, "a" becomes "2".
- If the character is in the third or fourth row, that character is encrypted by its row index followed by its column index. So "x" gets encrypted as "35".

The overall ciphertext is just the concatenation of the encryptions of each letter. So for example, "attack." is encrypted to "2 4 4 2 30 5 33". The spaces between the numbers are just to show how the various letters map to numbers; in reality the actually ciphertext would be "244230533" with no spaces.

(a) Explain how to decrypt. Explain for general keys, not just the key given above.

(b) What is the size of the key space?

(c) Suppose that we changed how we encrypt characters in the third and fourth row to be column index followed by row index. So for example now "x" becomes "53". Demonstrate that decryption would then be impossible: there will be two plaintexts that map to the same ciphertext. You may use the concrete example above to generate your ciphertexts.

(d) A friend suggests putting characters such as "a", "e", "i", "o", "r", "t" and maybe "_" in the second row. What non-security reasons might your friend have for this suggestion? What are some security trade-offs, compared to a random placement of characters.