Assignment #2 (now complete)
Submit electronically in PDF form by 11:59PM, November 11 (note change), with subject
CSC2426 to
jaiganesh at cs.toronto.edu
The file name should be: firstName LastName student# a2
and pdf files should be letter-sized 8.5X11 inches

1. (10 marks) Let $F$ be a function generator where for each $k \in \{0, 1\}^n$, $F_k : \{0, 1\}^* \rightarrow \{0, 1\}^n$. Assume that $F$ is pseudo-random with respect to adversaries that are restricted so that for every $n$, all of the queries to a function $f : \{0, 1\}^* \rightarrow \{0, 1\}^n$ are of the same length.

Define the function generator $F'$ as follows:
For $k \in \{0, 1\}^n$, $F'_k$ will map $\{0, 1\}^*$ to $\{0, 1\}^n$ as follows:
let $x \in \{0, 1\}^*$; then define
$F'_k(x) = F_k(\ell) = F_k(\ell \oplus CBC_k(x))$ where $\ell$ is the $n$-bit representation of the length of $x$.

Prove that $F'$ is pseudo-random (against unrestricted adversaries).

HINT: As usual, you should use your intuition about why this is true to help you find the proper hybrids. You may find it helpful to prove a lemma that, informally says the following:

If an (unrestricted, in the sense of being allowed to query strings of different lengths) adversary can distinguish between the following two situations:

- $k_0, k_1, k_2, \ldots$ are randomly chosen from $\{0, 1\}^n$, and when the adversary queries his input function on an $\alpha \in \{0, 1\}^i$, then the value $F_{k_i}(\alpha)$ is returned;

- a random function $f$ is chosen, and when he queries his input function on an $\alpha \in \{0, 1\}^*$, then the value $f(\alpha)$ is returned.

then there is a restricted adversary that breaks the pseudo-randomness of $F$.

2. (10 marks) Let $F$ be a function generator where for each $k \in \{0, 1\}^n$, $F_k : \{0, 1\}^n \rightarrow \{0, 1\}^n$. Let $CBC$ be defined from $F$ as in “Construction 2” on Page 2 of Notes #3; for $k \in \{0, 1\}^n$, $CBC_k : (\{0, 1\}^n)^* \rightarrow \{0, 1\}^n$. If $F$ is pseudo-random, then we know that $CBC$ is pseudo-random against adversaries that are restricted so that for every $n$, all of the queries to a function $f : \{0, 1\}^* \rightarrow \{0, 1\}^n$ are of the same length. The notes claim that “prepending the length” is a good way of getting rid of this restriction. For this question, you will prove that “postpending” the length is a very bad way.

For $k \in \{0, 1\}^n$, define $F'_k : (\{0, 1\}^n)^* \rightarrow \{0, 1\}^n$ as follows:
let $x \in (\{0, 1\}^n)^*$; then define
$F'_k(x) = CBC_k(x\ell) = F_k(\ell \oplus CBC_k(x))$ where $\ell$ is the $n$-bit representation of $|x|/n$. 

1
Prove that $F'$ is not pseudo-random against unrestricted adversaries, no matter what $F$ is.

**HINT:** Given a function $f$ for security parameter $n$, query $f$ on inputs of length $0$ and/or $n$, and/or $2n$.

3. (10 marks) Prove the Theorem towards the top of Page 3 of Notes #3 that states how to combine a privately collision resistant hash family and a (normal) pseudo-random function generator to create a pseudo-random function generator with unbounded inputs.

4. (30 marks)
The goal of this question is to show, as was suggested by a student in class, that one can get a secure session protocol by doing authentication and then encrypting for privacy.

The protocol will use function generators $F$ and $G$ where, for $|k| = n$, $F_k : \{0,1\}^{2n} \rightarrow \{0,1\}^n$ and and $G_k : \{0,1\}^n \rightarrow \{0,1\}^{2n}$. The key will consist of two parts: $k_1$ and $k_2$ where $|k_1| = |k_2| = n$.

Encryption works as follows:
The input pieces are of length $n$; pieces $m_0, m_1, \ldots$ are encrypted as $e_0, e_1, \ldots$ where

$$e_i = [m_i F_{k_1}(\overline{i}m_i)] \oplus G_{k_2}(\overline{i})$$

By $\overline{i}$ we mean the $n$-bit string representing integer $i \pmod{2^n}$. The security of this is much less obvious than if we had encrypted for privacy and then authenticated. Integrity is reasonably easy, as we will show in this question, since it only depends on the security (that is pseudo-randomness) of $F$. As we will see in the next assignment, privacy actually requires the security of (both) $F$ and $G$!

a) (5 marks) Describe carefully how $B$ decrypts. (If you do not do this correctly, then you cannot prove security of the protocol correctly!)

b) (10 marks) Prove that if $F$ is pseudo-random, then the given system satisfies integrity.

**Hint:** Given a distinguisher that breaks the integrity of this protocol, in order to break the pseudo-randomness of $F$, begin by choosing $k_2$. 
