Homework 4 for 6.875

Teacher: Silvio Micali, TA: Yevgeniy Dodis

Due Date: **Wednesday, November 4**

Problem 1: PRGs and Permutations

Let p be a polynomial, let $G : \{0,1\}^k \mapsto \{0,1\}^{p(k)}$ be a pseudo-random generator, and let $h : \{0,1\}^* \mapsto \{0,1\}^*$ be a function that is a permutation over $\{0,1\}^j$ when restricted to $\{0,1\}^j$, for all $j > 0$.

(a) Is it necessarily true that the distribution $h(G(s))$ (where s is chosen uniformly at random from $\{0,1\}^k$) is polynomially indistinguishable from the uniform distribution over $\{0,1\}^{p(k)}$? Is $h(G(s))$ necessarily a pseudo-random generator?

(b) Is it necessarily true that the distribution $G(h(s))$ (where s is chosen uniformly at random from $\{0,1\}^k$) is polynomially indistinguishable from the uniform distribution over $\{0,1\}^{p(k)}$? Is $G(h(s))$ necessarily a pseudo-random generator?

(c) Will any of yours answers to Parts (a) or (b) change if it is known that h is polynomial-time computable?

Problem 2: Next Bit Unpredictability $\Longrightarrow$ Passing all Statistical Tests

In class we proved that if the output of a generator $G : \{0,1\}^k \mapsto \{0,1\}^n$ (here n is some polynomial of k) passes the next bit unpredictability test, then it passes any statistical test. The proof used a hybrid argument to show that if there was a polynomial time statistical test A that distinguishes a completely random string from one generated by G, then the test could distinguish between a string in which the first i bits are from G and the rest random, and a string in which the first $i+1$ bits are from G and the rest random. To complete the proof, we then need to show how to use this to predict the next bit $(i+1)$ from the first i bits with probability non-negligibly better than $\frac{1}{2}$. Below are some suggestions of how to produce such a guess for the $(i+1)$st bit.

For each of the suggested predictors, give a convincing explanation of whether it is indeed a good predictor or not. Supply a formal proof for *one* of the good predictors. That is, prove that it indeed guesses correctly with probability better than $\frac{1}{2} + \frac{1}{Q(k)}$ for some polynomial Q. Denote by G_m the first m bits of $G(x)$ (where x is a random seed), and by R_m (or R'_m) a sequence of m random bit chosen from the uniform distribution. Assume without loss of generality that $\Pr[A(G_i R_{n-i}) = 0] = p$, and that $\Pr[A(G_{i+1} R_{n-i-1}) = 0] = p + \frac{1}{k^c}$ for some $c > 0$ (that is, we are assuming w.l.o.g. that A outputs 0 more often when the $(i+1)$st bit is from G). We are now given i bits G_i, and want to guess the next bit. Consider the following predictors.

(a) Run the test A first on $G_i 0 R_{n-i-1}$ and call the output a_0. Then run A on $G_i 1 R'_{n-i-1}$ and call the output a_1. If $a_0 = a_1$ output 0, otherwise output 1.

(b) Run the test A first on $G_i 0 R_{n-i-1}$ and call the output a_0. Then run A on $G_i 1 R'_{n-i-1}$ and call the output a_1. If $a_0 = a_1$ choose the output to be 0 or 1 randomly (with probability $\frac{1}{2}$). Otherwise, output the bit b for which $a_b = 0$ (that is, if $a_0 = 0$ output 0, and if $a_1 = 0$ output 1).

(c) Run the test A on $G_i R_{n-i}$. If the answer is 0, output the first bit of R_{n-i} (which is the $(i+1)$st bit in the string above). If the answer is 1, output the negation of that bit.

(d) Run the test A on $G_i 0 R_{n-i-1}$ for polynomially many times (each time with new independent R_{n-i-1}), and count how many times A outputs 0. If this fraction is closer to $p + \frac{1}{k^c}$ than to p, then output 0, otherwise output 1.

Problem 3: PRG $\Longrightarrow$ OWF

Let G be a pseudo-random generator. Show that G implies the existence of a length preserving one way function f. That is, construct a length preserving f using G, and prove that f is one way, using the fact that G is polynomially unpredictable.

Problem 4: Family of Trapdoor Permutations $\Longrightarrow$ GM-secure encryption scheme

Show that the existence of a family of trapdoor permutations implies the existence of a GM-secure (e.g. under three-pass) encryption scheme.