Cryptography Homework 2

陈贲(12212231)

Problem 1. In the following, $g(n)$ and $h(n)$ are both negligible functions, and p is a polynomially-bounded function. For each function below, decide if the function is guaranteed to be negligible or not. If so, prove it. If not, provide an example of negligible g and h and polynomially bounded p such that the function below is not negligible.

Solution.

(a) Negligible. Proof: W.l.o.g., assume $g(n) \leq h(n)$. Then $f_a(n) = h(n)$, which is negligible.

(b) Not negligible. Example: Since $g(n)$, by definition, $g(n) < 1/p(n)$ and $f_b(n) = 1/p(n)$. Suppose $p(n) = n/2 < n$, which is polynomially bounded. Then $f_b(n) = 1/p(n) = 2/n$, which is not negligible.

(c) Negligible. Proof: W.l.o.g., assume $g(n) \leq h(n)$. Then $f_c(n) = g(n)$, which is negligible.

(d) Negligible. Proof: By definition, $g(n) < 1/p(n)$, then $f_d(n) = g(n)$ which is negligible.

(e) Negligible. Proof: Since $g(n)$ and $h(n)$ are negligible, forall polynomial $p(n)$, there exists integers N_1 and N_2 such that $\forall n > N_1, g(n) < 1/p(n)$ and $\forall n > N_2, h(n) < 1/p(n)$. Let $N = \max(N_1, N_2)$. Then $\forall n > N, f_e(n) = g(n) + h(n) < 2/p(n)$, which is negligible.

(f) Negligible. Proof: Same as (e), $\forall n > N, f_f(n) = g(n)h(n) < (1/p(n))^2 < p'(n)$, which is negligible.

(g) Negligible. Proof: Since $g(n)$ is negligible, $\forall p(n)$, there exists integer N such that $\forall n > N, g(n) < 1/p(n)$. Then it still holds that $g(n) < 1/p^2(n)$ for some integer N' since $p^2(n)$ is polynomial. Thus, $\forall n > N', g(n)^{1/2} < p(n)$, which is negligible.

(h) Not negligible. Example: Suppose $g(n) = 1/n^{\log n}$ which is a negligible function, then $f_h(n) = 1/n$ which is not negligible.

Problem 2. Prove that property of computational indistinguishability: if $X_n \approx Y_n$ and f is a polynomial-time computable function, then $f(X_n) \approx f(Y_n)$.

Solution. Since for every polynomial-time algorithm A and large number n, we have $X_n \approx Y_n$, by definition, there exists a negligible function ε such that

$$|\Pr[A(X_n) = 1] - \Pr[A(Y_n) = 1]| < \varepsilon(n)$$

Suppose we have a new algorithm A' that will compute $f(X_n)$ and feed it into algorithm A. Then since f and A is polynomial-time computable, A' becomes polynomial-time algorithm, which means

$$|\Pr[A'(X_n) = 1] - \Pr[A'(Y_n) = 1]| < \varepsilon'(n)$$

and it is equivalent to

$$|\Pr[A(f(X_n)) = 1] - \Pr[A(f(Y_n)) = 1]| < \varepsilon'(n)$$

Thus, $f(X_n) \approx f(Y_n)$.

Problem 3. Prove that if (Gen, Enc, Dec) is a computationally secure encryption with $\ell(n)$-long messages, then for every polynomial-time algorithm Eve and large enough n, the probability that Eve wins in the following game is smaller than 0.34:

1. Eve gets as inputs 1^n, and gives Alice three strings $x_0, x_1, x_2 \in \{0,1\}^{\ell(n)}$
2. Alice chooses a random key $k \leftarrow_R \{0,1\}^n$ and $i \leftarrow_R \{0,1,2\}$ and computes $y = Enc_k(x_i)$
3. Eve gets y and outputs $j \in \{0,1,2\}$
4. Eve wins if $j = i$

Solution. Since the scheme is compuattionally secure, by definition, it holds that

$$\left|\Pr\left[A\left(E_{U_n}(x_i)\right) = j\right] - \Pr\left[A\left(E_{U_n}(x_j)\right) = k\right]\right| \leq \varepsilon(n)$$

where $i, j, k \in \{0,1,2\}$. Specifically, we have for $k \neq j$,

$$\Pr\left[A\left(E_{U_n}(x_k)\right) = j\right] \geq \Pr\left[A\left(E_{U_n}(x_j)\right) = j\right] - \varepsilon$$

then we have

$$\begin{aligned} 3 &= \sum_{i,j \in \{0,1,2\}} \Pr\left[A\left(E_{U_n}(x_i)\right) = j\right] \\ &\geq 3\sum_{i=0}^{2} \Pr\left[A\left(E_{U_n}(x_i)\right) = i\right] - 3\varepsilon(n) \end{aligned}$$

which gives that for large number n,

$$\frac{1}{3}\sum_{i=0}^{2} \Pr\left[A\left(E_{U_n}(x_i)\right) = i\right] \leq \frac{1}{3} + \frac{1}{3}\varepsilon(n) < 0.34$$

Problem 4. A sequence $\{X_n\}_{n\in\mathbb{N}}$ of distributions is pseudorandom if it's computationally indistinguishable from the sequence $\{U_n\}$ where U_n is the uniform distribution over $\{0,1\}^n$. Are the following sequences pseudorandom? Prove or refute it.

Solution. (1) No. Proof: We can construct a algorithm A that can distinguish the two sequences. For a sequence $\{x_i\}$, check that $x_n = x_1 \oplus x_2 \oplus ... \oplus x_{n-1}$, and the boolean expression gives the output of A. So, we have

$$|\Pr[A(X_n) = 1] - \Pr[A(U_n) = 1]| = \left|1 - \frac{1}{2}\right| = \frac{1}{2}$$

which means we can distinguish it from the uniform distribution and it's not pseudorandom.

(2) Yes. Proof: For small n, Z_n has the same distribution as U_n, and for large n, Z_n has the probability of $1 - 2^{-n/10}$ to be the same as U_n. Thus, for every algorithm A, it can be distinguished only when the plaintext appears and we have

$$|\Pr[A(Z_n) = 1] - \Pr[A(U_n) = 1]| = \left|\frac{1}{2}\left(1 - 2^{-n/10}\right) + 2^{-n/10} - \frac{1}{2}\right| < 2^{-n/10}$$

since $2^{-n/10}$ is negligible, the sequences are pseudorandom.

Problem 5. Let G be a pseudorandom generator where $|G(s)| > 2|s|$. Take $s = s_1...s_n$, and for simplicity, n is even.

(1) Define $G'(s) := G\big(s0^{|s|}\big)$. Is G' nessessarily a pseudorandom generator?
(2) Define $G'(s) := G\Big(s_1...s_{n/2}\Big)$, where $s = s_1...s_n$. Is G' nessessarily a pseudorandom generator?

Solution. (1) No. Proof: By contradiction, if G' is a PRG, then $G\big(0^{|s|}\big)$ will also be a PRG. However, we can construct a distinguisher D such that with random input i, it evaluates to 1 if $r = G(0^n)$ where $|i| = \ell(n)$. The distinguisher succeed with probability $1 - 2^{-|r|}$ which is not negligible and contrary to the assumption. Thus, G' is not PRG.

(2) Yes. Proof: Suppose for a probabilistic polynomial-time distinguisher D, we define

$$\varepsilon'(n) := \Big|\Pr_{r\leftarrow\{0,1\}^{\ell(n)}}[D(r) = 1] - \Pr_{s\leftarrow\{0,1\}^n}[D(G'(s)) = 1]\Big|$$

From the definition of G', for a seed s' with the first half of s and zeros for the rest $n/2$, then

$$\Pr_{s\leftarrow\{0,1\}^{n/2}}[D(G'(s)) = 1] = \Pr_{s\leftarrow\{0,1\}^{n/2}}[D(G(s')) = 1]$$

since G is a PRG, specifically, we have

$$\Big|\Pr_{r\leftarrow\{0,1\}^{\ell(n)}}[D(r) = 1] - \Pr_{s\leftarrow\{0,1\}^{n/2}}\Big[D\Big(G\Big(s_1...s_{n/2}\cdot 0^{n/2}\Big)\Big) = 1\Big]\Big| \le \varepsilon(n/2)$$

substituting the above equation, we have

$$\varepsilon(n/2) = \varepsilon'(n) \Rightarrow \varepsilon' \text{ is negligible function}$$

Thus, by the definition of PRG, G' is a PRG.

Problem 6. Define the keyed, length-preserving function F_k by $F_k(x) = F(k, x) = k \oplus x$. It's known that for any input x, the value of $F_k(x)$ is uniformly distributed if k is uniformly chosen. Prove of disprove that F_k is a PRF or not.

Solution. No. Proof: By definition, we can construct a distinguisher D and a random oracle $\mathcal{O}$: for two distinct values x_1, x_2 generated by F_k, the oracle obtains two values $y_1 = \mathcal{O}(x_1)$ and $y_2 = \mathcal{O}(x_2)$. The D will evaluate to 1 if it finds $F_k = k \oplus x$, i.e., $y_1 \oplus y_2 = x_1 \oplus x_2$. If $\mathcal{O} = F_k$, D evaluates to 1, otherwise, $\mathcal{O} = f$ where f is a uniformly random function, D evaluates to 1 when $f(x_1) \oplus f(x_2) = x_1 \oplus x_2$, which is equivalent to $f(x_1) = x_1 \oplus x_2 \oplus f(x_2)$ with probability 2^{-n}, by definition, $|1 - 2^{-n}|$ is not negligible, thus F_k is not a PRF.

Problem 7. Prove that if F_k is a length-preserving PRF, then

$$G(S) = F_s(\langle 1\rangle) \mid F_s(\langle 2\rangle) \mid ... \mid F_s(\langle \ell\rangle)$$

is a PRG with expansion factor $\ell \cdot n$, where $\langle i\rangle$ denotes the n-bit binary expression of the integer i.

Solution. Proof: Suppose G is not a PRG, then there exists a distinguisher D that can distinguish G from uniformly random string with non-negligible probability and an oracle. So, we can construct

an adversary D' that can distinguish F_k with the follwoing strategy: takes 1s as input, and queries the oracle ℓ times to compute $y = \mathcal{O}(\langle 1\rangle) \mid \mathcal{O}(\langle 2\rangle) \mid \cdots \mid \mathcal{O}(\langle \ell\rangle)$. Then D' evaluates to 1 if $D(y) = 1$, otherwise 0. By definition, since D is a PPT distinguisher, we have

$$
\begin{aligned}
&\left|\Pr_{s\leftarrow\{0,1\}^n}\left[D'^{F_s(\cdot)}(1^n)=1\right] - \Pr_{s\leftarrow\{0,1\}^n}\left[D'^{f(\cdot)}(1^n)=1\right]\right| \\
&= \left|\Pr_{s\leftarrow\{0,1\}^n}\left[D\Big(F_{s(\langle 1\rangle)} \mid \cdots \mid F_s(\langle \ell\rangle)\Big)=1\right] - \Pr_{f\leftarrow \mathrm{Func}_n}\left[D\Big(F_s(\langle 1\rangle) \mid \cdots \mid F_{s(\langle \ell\rangle)}\Big)=1\right]\right| \\
&= \left|\Pr_{s\leftarrow\{0,1\}^n}[D(G(s))=1] - \Pr_{r_i\leftarrow\{0,1\}^n}[D(r_1 \mid \cdots \mid r_\ell)=1]\right| \\
&= \left|\Pr_{s\leftarrow\{0,1\}^n}[D(G(s))=1] - \Pr_{r\leftarrow\{0,1\}^{\ell\cdot n}}[D(r)=1]\right| \geq \varepsilon(n)
\end{aligned}
$$

the adversary D' is a efficient distinguisher, and thus F_k is not a PRF. Therefore, by reduction, if F_k is a PRF, then G is a PRG.

Problem 8. Consider a variant of CBC-mode encryption where the sender simply increments the IV by 1 each time a message is encrypted (rather than choosing IV at random). Show that the resulting scheme is not CPA-secure.

Solution. By the definition of CPA security, suppose we have two fixed messages $m_0 = 0^n, m_1 \leftarrow \{0,1\}^n$. We can construct a algorithm D with non-negligible probability to win: query the encryption oracle to get $\mathcal{O}(0^{n-1}1) = \langle IV, c\rangle$. If the IV is even, then $IV + 1 = IV \oplus 0^{n-1}1$, and therefore, $c = F_k(IV \oplus 0^{n-1}1) = F_k(IV + 1)$. Suppose the challenge ciphertext is c', if $c = c'$, D outputs 0 since $F_k(IV + 1) = F_k((IV + 1) \oplus 0^n)$. If the IV is odd, the algorithm outputs a random bit and has $1/2$ chance to win. The algorithm D has non-negligible probability to win, and thus the scheme is not CPA-secure.

Problem 9. Consider a variant of CBC-mode encryption called chained CBC mode, where the last block of the previous ciphertext is used as the IV for the next message. It reduced the bandwidth, since the IV need not be sent. An initial message m_1, m_2, m_3 is encrypted using a random IV and subsequent messages m_4, m_5 is encrypted using c_3 as the IV. However, the chained CBC mode is not as secure as CBC mode. Show a chosen-plaintext attack scheme.

Solution. Suppose the attacker knows the messages $m_1 \in \{m_0^1, m_1^1\}$, after getting the first responces $\langle IV, c_1, c_2, c_3\rangle$, the attacker requests the encryption oracle to encrypt $m_4 = IV \oplus m_0^1 \oplus c_3$ and $m_5 \leftarrow \{0,1\}^n$. The attacker can determine that if $c_4 = c_1$, then $m_1 = m_0^1$