Homework 12

Problem 1. Let G be a pseudorandom generator of stretch ℓ such that $\ell(n) \geq 2n$.

(a) Define G' as $G'(s) = G(s0^{|s|})$. Is G' necessarily a pseudorandom generator?

(b) Define G'' as $G''(s) = G(s_1 \cdots s_{n/2})$ for $s = s_1 s_2 \cdots s_n$. Is G'' necessarily a pseudorandom generator?

Solution. (a) No. We will give a counterexample below. Consider another PRG H with stretch ℓ, and define G as: on input x with length $|x| = 2n$:

$$G(x) = \begin{cases} 0^{\ell(2n)} & \text{if } x_{n+1}, ..., x_{2n} = 0 \\ H(x) & \text{else} \end{cases}$$

We will now prove that this G is a PRG. We have:

$$\begin{aligned}
& \Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(G(x)) = 1\Big) \\
= & \Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(G(x)) = 1\Big| x_{n+1}, ..., x_{2n} = 0\Big) \Pr\Big(x_{n+1}, ..., x_{2n} = 0\Big) \\
& + \Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(G(x)) = 1\Big| x_{n+1}, ..., x_{2n} \text{ not all } 0\Big) \Pr\Big(x_{n+1}, ..., x_{2n} \text{ not all } 0\Big) \\
= & \Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(0) = 1\Big| x_{n+1}, ..., x_{2n} = 0\Big) \Pr\Big(x_{n+1}, ..., x_{2n} = 0\Big) \\
& + \Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(H(x)) = 1\Big| x_{n+1}, ..., x_{2n} \text{ not all } 0\Big) \Pr\Big(x_{n+1}, ..., x_{2n} \text{ not all } 0\Big) \\
= & \Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(0) = 1\Big) \times 2^{-n} + \Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(H(x)) = 1\Big) \times (1 - 2^{-n})
\end{aligned}$$

Thus:

$$
\begin{aligned}
&\left|\Pr_{s\in\{0,1\}^{2n}}\Big(\mathcal{A}(G(x))=1\Big)-\Pr_{r\in\{0,1\}^{\ell(2n)}}\Big(\mathcal{A}(r)=1\Big)\right|\\
=&\left|\Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(0)=1\Big)\times 2^{-n}+\Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(H(x))=1\Big)\times(1-2^{-n})\right.\\
&\left.-\Pr_{r\in\{0,1\}^{\ell(2n)}}\Big(\mathcal{A}(r)=1\Big)\right|\\
\leq&(1-2^{-n})\left|\Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(H(x))=1\Big)-\Pr_{r\in\{0,1\}^{\ell(2n)}}\Big(\mathcal{A}(r)=1\Big)\right|\\
&+2^{-n}\left|\Pr_{x\in\{0,1\}^{2n}}\Big(\mathcal{A}(0)=1\Big)-\Pr_{r\in\{0,1\}^{\ell(2n)}}\Big(\mathcal{A}(r)=1\Big)\right|\\
\leq&(1-2^{-n})\text{negl}(2n)+2^{-n}\times 1\\
=&\text{negl}(2n)
\end{aligned}
$$

Then consider G' with this G. We have $G'(s)=G(s0^{|s|})=0^{\ell 2n}$ by definition, so for any input s the encoding of G' is always constant, which means G' is not a PRG (we can set $\mathcal{A}$ to output 1 if and only if the input is all 0).

(b) Yes. Since $G''(s)=G(s_1\cdots s_{\frac{n}{2}})$ only depends on the first half of input s, and is independent with $s_{\frac{n}{2}+1},...,s_n$, so we have:

$$
\begin{aligned}
&\Pr_{s\in\{0,1\}^n}\Big(\mathcal{A}(G''(s))=1\Big)=\Pr_{s_1,s_2,...,s_n\in\{0,1\}}\Big(\mathcal{A}(G(s_1\cdots s_{\frac{n}{2}}))=1\Big)\\
=&\sum_{s_{\frac{n}{2}+1},...,s_n\in\{0,1\}}\Pr_{s_1,...,s_{\frac{n}{2}}\in\{0,1\}}\Big(\mathcal{A}(G(s_1\cdots s_{\frac{n}{2}}))=1\Big|s_{\frac{n}{2}+1}\cdots s_n\Big)\Pr(s_{\frac{n}{2}+1},...,s_n)\\
=&\Pr_{s_1,...,s_{\frac{n}{2}}\in\{0,1\}}\Big(\mathcal{A}(G(s_1\cdots s_{\frac{n}{2}}))=1\Big)\sum_{s_{\frac{n}{2}+1},...,s_n\in\{0,1\}}\Pr(s_{\frac{n}{2}+1},...,s_n)\\
=&\Pr_{s_1,...,s_{\frac{n}{2}}\in\{0,1\}}\Big(\mathcal{A}(G(s_1\cdots s_{\frac{n}{2}}))=1\Big)
\end{aligned}
$$

Meanwhile, as $|G''(s)|=|G(s_1\cdots s_{\frac{n}{2}})|=\ell(\frac{n}{2})=\ell''(n)$, so we have:

$$
\begin{aligned}
&\left|\Pr_{s\in\{0,1\}^n}\Big(\mathcal{A}(G''(s))=1\Big)-\Pr_{r\in\{0,1\}^{\ell''(n)}}\Big(\mathcal{A}(r)=1\Big)\right|\\
=&\left|\Pr_{s_1,...,s_{\frac{n}{2}}\in\{0,1\}}\Big(\mathcal{A}(G(s_1\cdots s_{\frac{n}{2}}))=1\Big)-\Pr_{r\in\{0,1\}^{\ell(\frac{n}{2})}}\Big(\mathcal{A}(r)=1\Big)\right|\\
=&\text{negl}(\frac{n}{2})\\
=&\text{negl}(n)
\end{aligned}
$$

The second last line is due to the fact that G is a PRG. Thus G'' is also a PRG.

Problem 2. A keyed family of functions F_k is a pseudorandom random permutation (PRP) if (a) $F_k(\cdot)$ and $F_k^{-1}(\cdot)$ are efficiently computable given the key k and (b) for any polynomial-time algorithm $\mathcal{A}$,

$$\left|\Pr\Big(\mathcal{A}^{F_k(\cdot),F_k^{-1}(\cdot)}(1^n)=1\Big)-\Pr\Big(\mathcal{A}^{f_n(\cdot),f_n^{-1}(\cdot)}(1^n)=1\Big)\right|\leq \mathrm{negl}(n).$$

Consider the following encryption scheme

1. Sample key k uniformly at random.
2. On input plaintext $x\in\{0,1\}^{n/2}$, algorithm Enc_k samples randomness $r\in\{0,1\}^{n/2}$ and outputs ciphertext $F_k(r\|x)$.

Solve the following problems assuming that F_k is a PRP.

(a) Show how the decryption Dec_k works.

(b) Prove that the encryption scheme is CPA-secure.

Solution. (a) Consider $\mathrm{Dec}_k: y\mapsto \Big(F_k^{-1}(y)\Big)[\frac{n}{2}+1:n]$ where "$[i:j]$" means taking the i-th to j-th bit. Then for any plaintext $x\in\{0,1\}^{n/2}$, we have $\mathrm{Dec}_k(\mathrm{Enc}_k(x))=\Big(F_k^{-1}(F_k(r\|x))\Big)[\frac{n}{2}+1:n]=(r\|x)[\frac{n}{2}+1:n]=x$.

(b) Consider $\widetilde{\Pi}=(\widetilde{\mathrm{Enc}},\widetilde{\mathrm{Dec}})$ where real random function f_n is used, and Π where F_k is used. If Π is not CPA, then there is some adversary $\mathcal{A}$ with the encryption oracle that can attack Π, which means $\Pr(A_\Pi \text{ succ})\geq 1/2+1/\mathrm{poly}(n)$.

Consider $\mathcal{A}_{\widetilde{\Pi}}$. It's success rate is strictly $1/2$ if it has not queried Enc using the same random r with the one used in the actual encryption. Meanwhile, the probability of the occurence of using the same random r is $q(n)/2^n$ where $q(n)$ is the number of queries $\mathcal{A}$ makes to Enc, which should be polynomial. Thus: $\Pr(\mathcal{A}_{\widetilde{\Pi}}\text{ succ})\leq 1/2+q(n)/2^n=1/2+\mathrm{negl}(n)$.

Using the above adversaries, We can construct a poly-time distinguisher D between (F_k,F_k^{-1}) and (f_n,f_n^{-1}) with oracles O_1,O_2 and input 1^n:

1. Run $\mathcal{A}(1^n)$, whenever encryption is called with input x, answer with $O_1(r\|x)$ where r is randomly sampled from $\{0,1\}^n$.

2. When $\mathcal{A}$ outputs x_0,x_1, choose random bit b, feed $O_1(r\|x_b)$ to $\mathcal{A}$ and get output b'.

3. Output $1_{b=b'}$.

Then, we know that $\Pr(D^{F_k,F_k^{-1}}(1^n) = 1) = \Pr(\mathcal{A}_\Pi \text{ succ})$, and $\Pr(D^{f_n,f_n^{-1}}(1^n) = 1) = \Pr(\mathcal{A}_{\widetilde{\Pi}} \text{ succ})$.

So:

$$\begin{aligned}
&\left|\Pr\left(D^{f_n,f_n^{-1}}(1^n) = 1\right) - \Pr\left(D^{F_k,F_k^{-1}}(1^n) = 1\right)\right| \\
=&\left|\Pr\left(\mathcal{A}_{\widetilde{\Pi}} \text{ succ}\right) - \Pr\left(\mathcal{A}_\Pi \text{ succ}\right)\right| \\
=&\frac{1}{\text{poly}(n)}.
\end{aligned}$$

which raises contradiction to the fact that F_k is PRP. Thus, the encryption scheme is CPA-secure.