Solution of Exercisesheet 9

Exercise 9-1

(a) Proof by reduction: Lets assume efficient adversary, A, against f, i.e, it breaks one wayness of the function i.e., A could invert the $f(x)$ with a non negligible probability. So A could find x' such that $f(x') = f(x)$. Because f is a PRP, f is bijective and therefore $x = x'$.

Now construct an adversary, A', against hardcore bit, h, using A. A' forwards any y he gets to A. With the recieved x' he then computes $hc(x')$ which he outputs.
A' invokes A and A is efficient. Therefore the message length of the messages must be poly. Forwarding the messages and computing $hc(x')$ can also be done in polynomial time. So A' is efficient.
A' simulates f perfectly to A, so the output of A is $x' = x$ with non negligible probability. A' then computes $hc(x')$ which is the same like $hc(x)$, because $x = x'$, so A' wins with the same non negligible probability.
But this is contradiction to our assumption, as for the hardcore bit it is not possible to find a value h which is equal to $hc(x)$ with a non negligible probability. Hence such A does not exist. Hence this contrunction is secure.

(b) Let f be a constant function and h be most significant bit, $msb(x)$. For this function it is hard for an Adversary to compute $h(x)$ from $f(x)$. Because $f(x) = c \forall x$ and this makes it impossible to know/compute x which is necessary to compute $hc(x)$. Constant function is not a one way function. Because for constant function any value from domain as input to f will be same as the recieved $f(x)$. Hence the above conclusion from (a) is not true for a OWF.

Exercise 9-2

We assume for the sake of contradiction an efficient inverter $\mathcal{A}$ for G that breaks G with non-negligible probability $\epsilon(\lambda)$ and then build a distinguisher $\mathcal{D}$ for G. This distinguisher gets a value y from an oracle which either outputs $G(s)$ ($G : \{0,1\}^k \to \{0,1\}^{2k}$) or a random value r of length $2k$. He forwards this value to the inner adversary $\mathcal{A}$, which outputs a value x', for which $G(x') = y$ holds, if the input is in the output space of G otherwise $\mathcal{A}$ outputs (nothing/ a fixed value)???. $\mathcal{D}$ can then distinguish, if the input y to $\mathcal{A}$ was in the output space of G, then he returns 0, otherwise the input y was a random message, then he returns 1.
$\mathcal{D}$ is efficient, because he invokes $\mathcal{A}$ and $\mathcal{A}$ is efficient. It also follows, that the message length of the messages $\mathcal{D}$ forwards has to be poly and forwarding is in poly time too.
If the oracle outputs a pseudorandom string $G(s)$, $\mathcal{D}$ perfectly simulates G to $\mathcal{A}$. It follows, that $Pr[\mathcal{D}(G(s)) = 1] = Pr[Invert_{\mathcal{A},G}(\lambda) = 1] = \epsilon(\lambda)$.
For the case where the oracle outputs a random string r, the probability that one message is in the output space of G is $\frac{2^k}{2^{2k}} = 2^{-k}$. (Here we need that the output length of G is $2k$ and not $k+1$.)
For q messages the probability is $\frac{q}{2^{-k}}$, but since q is poly (because $\mathcal{A}$ has to be efficient), this is still negligible.
So $|Pr[\mathcal{D}(G(s)) = 1] - Pr[\mathcal{D}(r) = 1]| = \epsilon(\lambda) - \frac{q}{2^{-k}} > negl(\lambda)$. This is a contradiction to the pseudorandomness of G, so such an adversary $\mathcal{A}$ cannot exist. It follows that $G(s)$ is a OWF by

itself.

Exercise 9-3

(a) **To show:** Prove that regular CPA security implies λ-CPA security.
We do this by a reduction. We assume there is an efficient adversary $\mathcal{A}$ against the λ-CPA-security of Π which is successful with non-negligible probability. From this we construct our adversary $\mathcal{B}$ against the CPA-security of Π which invokes $\mathcal{A}$. $\mathcal{B}$ has to provide an encryption oracle for $\mathcal{A}$. To do this, he forwards any message m $\mathcal{A}$ sends to his oracle to his own oracle and recieves the ciphertext c. He then makes a vector $\vec{C}$, which contains λ-times the ciphertext c, and forwards it to $\mathcal{A}$.
$\mathcal{A}$ eventually outputs two messages $(\widetilde{m_0}, \widetilde{m_1})$, which $\mathcal{B}$ forwards to his challenger. Then he sends an vector $\vec{C_b}$ to $\mathcal{A}$, which contains λ-times the recieved ciphertext c_b. Then $\mathcal{B}$ outputs the same bit b like $\mathcal{A}$ does.
$\mathcal{B}$ invokes $\mathcal{A}$ and $\mathcal{A}$ is efficient. Because of that, the message length have to be poly. Furthermore forwarding messages is in poly time too. So $\mathcal{B}$ is efficient.
To analyse the success, we ascertain, that $\mathcal{B}$ simulates the λ-CPA-game perfectly to $\mathcal{A}$. So the success probability of $\mathcal{B}$ is the same as $\mathcal{A}$, which is non-negligible. This is a contradiction to the CPA security of $\mathcal{B}$, so such an adversary $\mathcal{A}$ cannot exit.
It follows that the scheme is λ-CPA secure, if it CPA secure. In other words, regular CPA security implies λ-CPA security.

(b) **To show:** Prove that λ-CPA security implies normal CPA security.
We do this by a reduction. We assume there is an efficient adversary $\mathcal{A}$ against the CPA-security of Π which is successful with non-negligible probability. From this we construct our adversary $\mathcal{B}$ against the λ-CPA-security of Π which invokes $\mathcal{A}$. $\mathcal{B}$ has to provide an encryption oracle for $\mathcal{A}$. To do this, he forwards any message m $\mathcal{A}$ sends to his oracle to his own oracle and recieves the ciphertextvector $\vec{C} = (c_1, ..., c_\lambda)$. He then forwards only the first ciphertext c_1 to $\mathcal{A}$.
$\mathcal{A}$ eventually outputs two messages $(\widetilde{m_0}, \widetilde{m_1})$, which $\mathcal{B}$ forwards to his challenger. From the recieved ciphertextvector $\vec{C_b}$ he again forwards only the first ciphertext to $\mathcal{A}$. Then $\mathcal{B}$ outputs the same bit b like $\mathcal{A}$ does.
$\mathcal{B}$ invokes $\mathcal{A}$ and $\mathcal{A}$ is efficient. Because of that, the message length have to be poly. Furthermore forwarding messages is in poly time too. So $\mathcal{B}$ is efficient.
To analyse the success, we ascertain, that $\mathcal{B}$ simulates the CPA-game perfectly to $\mathcal{A}$. So the success probability of $\mathcal{B}$ is the same as $\mathcal{A}$, which is non-negligible. This is a contradiction to the λ-CPA security of $\mathcal{B}$, so such an adversary $\mathcal{A}$ cannot exit.
It follows that the scheme is CPA secure, if it λ-CPA secure. In other words, λ-CPA security implies normal CPA security.

Exercise 9-4

(a) Prove that any PRF is also a (t-keys) PRF for all choices of $t = \text{poly}(\lambda)$
We assume there is an efficient adversary $\mathcal{A}$ against a (t-keys) PRF which manages to dis-

tinguish a PRF against a random function with non-negligible probability. We construct the distinguisher $\mathcal{D}$ against a PRF which invokes $\mathcal{A}$.
$\mathcal{D}$ answers queries from $\mathcal{A}$ to either the PRF or a random function and recieves the result $y = F(k, \cdot)$ or $f(\cdot)$. For each result $\mathcal{D}$ creates a vector $\vec{V}$, which contains t-times y and forwards it to $\mathcal{A}$.
$\mathcal{A}$ has to decide whether the recieved vector $\vec{V}$ contains $(y_1 = F(k_1, \cdot), \ldots, y_\lambda = F(k_\lambda, \cdot))$ or $(y_1 = f_1(\cdot), \ldots, y_\lambda = f_\lambda(\cdot))$. $\mathcal{A}$ displays its decision with bit b. $b = 0$ means PRF and $b = 1$ means the vector contains results of a truly random function. $\mathcal{D}$ outputs the same bit b as $\mathcal{A}$.
$\mathcal{D}$ invokes $\mathcal{A}$ and $\mathcal{A}$ is efficient. Therefore the message length of the messages to the query must be poly. Forwarding these queries is efficient and creating a vector of t-times the result of the queries y is poly since $t = \text{poly}(\lambda)$. So $\mathcal{D}$ is efficient.
To analyse the success, $\mathcal{D}$ simulates a (t-keys) PRF perfectly to $\mathcal{A}$. So the success probability of $\mathcal{D}$ is the same as $\mathcal{A}$, which is non-negligible. This is a contradiction to the PRF security of $\mathcal{B}$, so such an adversary $\mathcal{A}$ cannot exit.

(b) Prove that for all choices of $t = \text{poly}(\lambda)$ and any (t-keys) PRF is also a PRF
We assume there is an efficient adversary $\mathcal{A}$ against a PRF manages to distinguish a (t-keys) PRF against a random function with non-negligible probability. We construct the distinguisher $\mathcal{D}$ against a (t-keys) PRF which invokes $\mathcal{A}$.
$\mathcal{D}$ answers queries from $\mathcal{A}$ to either the (t-keys) PRF or a random function and recieves the result vector $\vec{V} = (y_1 = F(k_1, \cdot), \ldots, y_\lambda = F(k_\lambda, \cdot))$ or $(y_1 = f_1(\cdot), \ldots, y_\lambda = f_\lambda(\cdot))$. $\mathcal{D}$ forwards the first result of $\vec{V}$ y_1 to $\mathcal{A}$.
$\mathcal{A}$ has to decide whether the recieved vector y_1 is the result of $F(k_1, \cdot)$ or $f_1(\cdot)$. $\mathcal{A}$ displays its decision with bit b. $b = 0$ means PRF and $b = 1$ means the vector contains results of a truly random function. $\mathcal{D}$ outputs the same bit b as $\mathcal{A}$.
$\mathcal{D}$ invokes $\mathcal{A}$ and $\mathcal{A}$ is efficient. Therefore the message length of the messages to the query must be poly. Forwarding these queries is efficient making $\mathcal{D}$ also efficient.
To analyse the success, $\mathcal{D}$ simulates a PRF perfectly to $\mathcal{A}$. So the success probability of $\mathcal{D}$ is the same as $\mathcal{A}$, which is non-negligible. This is a contradiction to the PRF security of $\mathcal{B}$, so such an adversary $\mathcal{A}$ cannot exit.