3 PRP versus Left-or-Right

WARNING: in this exercise, the definitions which are proposed are not correct. Instead of saying there is a negligible function which major the advantage of any *adversary, we should have said any adversary has a negligible advantage.*

Given a security parameter (which is implicit and omitted from notations for better readability), we consider a pair $(\mathsf{Enc}, \mathsf{Dec})$ of functions from $\{0,1\}^k \times \{0,1\}^n$ to $\{0,1\}^n$ (k and n are functions of the security parameter). These functions are such that for all K and X, we have

$$\mathsf{Dec}(K, \mathsf{Enc}(K, X)) = X$$

It is assumed that there are implementations which can evaluate both functions in polynomial time complexity (in terms of the security parameter). We define several security notions.

PRP. We say that this pair is a *pseudorandom permutation* (PRP) if there exists a negligible function negl such that for all probabilistic polynomial time (PPT) algorithm $\mathcal{A}$, we have $\Pr[\Gamma^{\mathsf{PRP}}(\mathcal{A}, 0) \to 1] - \Pr[\Gamma^{\mathsf{PRP}}(\mathcal{A}, 1) \to 1] \leq \mathsf{negl}$, where $\Gamma^{\mathsf{PRP}}(\mathcal{A}, b)$ is the PRP game defined as follows:

$\Gamma^{\mathsf{PRP}}(\mathcal{A}, b)$:
1: initialize a list $\mathcal{L}$ to empty
2: pick $K \in \{0,1\}^k$ uniformly at random
3: pick a permutation Π over $\{0,1\}^n$ uniformly at random
4: run $b' \leftarrow \mathcal{A}^{\mathcal{O}}$
5: return b'

subroutine $\mathcal{O}(x)$:
6: if $x \in \mathcal{L}$ abort
7: insert x in $\mathcal{L}$
8: **if** $b = 0$ **then**
9: return $\mathsf{Enc}(K, x)$
10: **else**
11: return $\Pi(x)$
12: **end if**

LoR. We say that this pair is *LoR-secure* if there exists a negligible function negl such that for all probabilistic polynomial time (PPT) algorithm $\mathcal{A}$, we have $\Pr[\Gamma^{\mathsf{LoR}}(\mathcal{A}, 0) \to 1] - \Pr[\Gamma^{\mathsf{LoR}}(\mathcal{A}, 1) \to 1] \leq \mathsf{negl}$, where $\Gamma^{\mathsf{LoR}}(\mathcal{A}, b)$ is the left-or-right game defined as follows:

$\Gamma^{\mathsf{LoR}}(\mathcal{A}, b)$:
1: initialize two lists $\mathcal{L}_l$ and $\mathcal{L}_r$ to empty
2: pick $K \in \{0,1\}^k$ uniformly at random
3: run $b' \leftarrow \mathcal{A}^{\mathcal{O}}$
4: return b'

subroutine $\mathcal{O}(x_l, x_r)$:
5: if $x_l \in \mathcal{L}_l$ or $x_r \in \mathcal{L}_r$, abort
6: insert x_l in $\mathcal{L}_l$ and x_r in $\mathcal{L}_r$
7: **if** $b = 0$ **then**
8: return $\mathsf{Enc}(K, x_l)$
9: **else**
10: return $\mathsf{Enc}(K, x_r)$
11: **end if**

We want to show the equivalence between these notions.

Q.1 Is the list management important in each security definition (or: what happens with modified definitions in which we remove the lists)? Justify your answer.

We consider the games $\Gamma^{\mathsf{PRP}*}$ and $\Gamma^{\mathsf{LoR}*}$ which are the same as Γ^{PRP} and Γ^{LoR}, respectively, without any list management or abort.
The list management is not important in the PRP security. Indeed, we could simulate a PRP$*$ *adversary $\mathcal{A}$ repeating queries by a* PRP *adversary $\mathcal{B}$ who does not repeat them, by simulating $\mathcal{A}$, remembering his queries and the responses, and simulating repeating queries instead of querying them. We would have* $\Pr[\Gamma^{\mathsf{PRP}*}(\mathcal{A}, b) \to 1] = \Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}, b) \to 1]$.
The list management is important in the LoR security. Indeed, the following adversary outputs b with probability 1 in a LoR$*$ *game in which repetitions are allowed:*
$\mathcal{A}^{\mathcal{O}}$*:*
1: *pick $x, y \in \{0,1\}^n$ such that $x \neq y$*
2: *query $u = \mathcal{O}(x, y)$*
3: *query $v = \mathcal{O}(y, y)$*
4: *answer $1_{u=v}$*
So, $\Pr[\Gamma^{\mathsf{LoR}*}(\mathcal{A}, 0) \to 1] - \Pr[\Gamma^{\mathsf{LoR}*}(\mathcal{A}, 1) \to 1] = 1$. *This is not negligible. So, no* LoR$*$ *security is feasible. Nevertheless, we will see that LoR and PRP are equivalent.*

Q.2 We consider the following hybrid game:
$\Gamma^{\mathsf{hyb}}(\mathcal{A}, b)$:
1: initialize a list $\mathcal{L}$ to empty
2: pick $K \in \{0,1\}^k$ uniformly at random
3: pick a permutation Π over $\{0,1\}^n$ uniformly at random
4: run $b' \leftarrow \mathcal{A}^{\mathcal{O}}$
5: return b'
subroutine $\mathcal{O}(x)$:
6: if $x \in \mathcal{L}$ abort
7: insert x in $\mathcal{L}$
8: **if** $b = 0$ **then**

9: return $\mathsf{Enc}(K, x)$
10: **else**
11: return $\mathsf{Enc}(K, \Pi(x))$
12: **end if**

Show that for all $\mathcal{A}$ playing the PRP game and any b, we have $\Pr[\Gamma^{\mathsf{PRP}}(\mathcal{A}, b) \to 1] = \Pr[\Gamma^{\mathsf{hyb}}(\mathcal{A}, b) \to 1]$.

For $b = 0$, the result is obvious: by getting rid of steps which are never executed, we can see that the two games are the same. So, we concentrate on $b = 1$.
If K is random and Π is an independent uniformly distributed permutation, then $\Pi'(x) = \mathsf{Enc}(K, \Pi(x))$ is also an independent uniformly distributed permutation. So, a bridging step in which we replace $\mathsf{Enc}(K, \Pi(x))$ by $\Pi'(x)$ with Π' selected randomly produces the same result.

Q.3 Given $\mathcal{A}$ playing the PRP game, we define $\mathcal{B}$ playing the LoR game as follows:
$\mathcal{B}^{\mathcal{O}}$:
1: pick a permutation Π over $\{0,1\}^n$ uniformly at random
2: run $\mathcal{A}$
when $\mathcal{A}$ makes a query x to its oracle, answer by $\mathcal{O}(x, \Pi(x))$
3: return the same output as $\mathcal{A}$

Show that $\Pr[\Gamma^{\mathsf{hyb}}(\mathcal{A}, b) \to 1] = \Pr[\Gamma^{\mathsf{LoR}}(\mathcal{B}, b) \to 1]$ for any b.

First of all, it is clear that some x repeats if and only if some $\Pi(x)$ repeats, because Π is a permutation. So, removing the $\mathcal{L}_r$ management in the LoR game with $\mathcal{B}$ does not change the outcome of the game. Then, the LoR game without $\mathcal{L}_r$ management can be changed into the hybrid game by bridging steps. So, $\Pr[\Gamma^{\mathsf{hyb}}(\mathcal{A}, b) \to 1] = \Pr[\Gamma^{\mathsf{LoR}}(\mathcal{B}, b) \to 1]$.

Q.4 Deduce that LoR-security implies PRP.
CAUTION: adversaries must be PPT.

The adversary $\mathcal{B}$ in the previous question is not polynomially bounded as it must pick a random Π. However, we can perfectly simulate it by using the lazy sampling technique: $\mathcal{B}'$ keeps a table of $(x, \Pi(x))$ pairs which is initially empty and, upon a new query x, checks if it is in the table, and if not, picks a random output y which is different than all previous ones, then insert (x, y) in the table.
If we have LoR security, given a (PPT) PRP adversary $\mathcal{A}$, the previous reduction makes a PPT adversary $\mathcal{B}'$ playing the LoR game and such that $\Pr[\Gamma^{\mathsf{PRP}}(\mathcal{A}, b) \to 1] = \Pr[\Gamma^{\mathsf{LoR}}(\mathcal{B}', b) \to 1]$ for any b. Since $\Pr[\Gamma^{\mathsf{LoR}}(\mathcal{B}', 0) \to 1] - \Pr[\Gamma^{\mathsf{LoR}}(\mathcal{B}', 1) \to 1] \leq \mathsf{negl}$, we have $\Pr[\Gamma^{\mathsf{PRP}}(\mathcal{A}, 0) \to 1] - \Pr[\Gamma^{\mathsf{PRP}}(\mathcal{A}, 1) \to 1] \leq \mathsf{negl}$. Since this holds for any PPT $\mathcal{A}$, we obtain PRP security.

Q.5 Using the following game, show that PRP security implies LoR security. Give a precise proof with the reductions.

$\Gamma^{\mathsf{generic}}(\mathcal{A}, b, c)$:

1: initialize two lists $\mathcal{L}_l$ and $\mathcal{L}_r$ to empty
2: pick $K \in \{0,1\}^k$ uniformly at random
3: pick a permutation Π over $\{0,1\}^n$ uniformly at random
4: run $b' \leftarrow \mathcal{A}^{\mathcal{O}}$
5: return b'

subroutine $\mathcal{O}(x_l, x_r)$:

6: if $x_l \in \mathcal{L}_l$ or $x_r \in \mathcal{L}_r$, abort
7: insert x_l in $\mathcal{L}_l$ and x_r in $\mathcal{L}_r$
8: **if** $b = 0$ **then**
9: **if** $c = 0$ **then**
10: return $\mathsf{Enc}(K, x_l)$
11: **else**
12: return $\Pi(x_l)$
13: **end if**
14: **else**
15: **if** $c = 0$ **then**
16: return $\mathsf{Enc}(K, x_r)$
17: **else**
18: return $\Pi(x_r)$
19: **end if**
20: **end if**

Let $\mathcal{A}$ be any (PPT) LoR adversary.
We have $\Pr[\Gamma^{\mathsf{LoR}}(\mathcal{A}, b) \to 1] = \Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, b, 0) \to 1]$.
When $c = 1$, as the queries never repeat, the oracle always returns a random answer which is different from all previous ones, no matter the value of b. So, we have $\Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, 0, 1) \to 1] = \Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, 1, 1) \to 1]$.
Using a bridging step, we construct $\mathcal{B}_b$ such that for all b and c, $\Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}_b, c) = \Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, b, c) \to 1]$. *So*

$$
\begin{aligned}
& \Pr[\Gamma^{\mathsf{LoR}}(\mathcal{A}, 0) \to 1] - \Pr[\Gamma^{\mathsf{LoR}}(\mathcal{A}, 1) \to 1] \\
&= \Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, 0, 0) \to 1] - \Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, 1, 0) \to 1] \\
&= \big(\Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, 0, 0) \to 1] - \Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, 0, 1) \to 1]\big) - \\
&\quad \big(\Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, 1, 0) \to 1] - \Pr[\Gamma^{\mathsf{generic}}(\mathcal{A}, 1, 1) \to 1]\big) \\
&= \big(\Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}_0, 0) \to 1] - \Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}_0, 1) \to 1]\big) - \\
&\quad \big(\Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}_1, 0) \to 1] - \Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}_1, 1) \to 1]\big) \\
&= \big(\Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}_0, 0) \to 1] - \Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}_0, 1) \to 1]\big) + \\
&\quad \big(\Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}'_1, 0) \to 1] - \Pr[\Gamma^{\mathsf{PRP}}(\mathcal{B}'_1, 1) \to 1]\big) \\
&\leq 2\mathsf{negl}
\end{aligned}
$$

where $\mathcal{B}'_b$ gives the opposite answer to $\mathcal{B}_b$. As $2\mathsf{negl}$ *is a negligible function, this is negligible. This applies to any $\mathcal{A}$. Hence, we have LoR security.*