Quantum Cryptography (spring 2022)	Dominique Unruh

Exercise Sheet 07

Out: 2022-04-25 Due: 2022-05-03

1 Missing claims from QKD proof

(a)

Knowlets:	RawKey ProblemID: CountTError
Time:	
Difficulty:	

In the practice we showed (or will show) that in our QKD protocol, after the Bell test and after measuring the n-bit raw key, we have

$$H_\infty(K_A|E)_{\rho_{raw}} \geq -\log(N2^{-n})$$

where $N := |\{xy \in \{0,1\}^{2n} : |xy| \leq t\}|$. (Note: $|xy|$ does not refer to the Hamming weight of xy here, but to the number of non-00 bitpairs.)

Show that $N \leq (3n+1)^t$.

Hint: Think of how you can compactly describe the bitstring xy with $|xy|$ by only telling where the non-00 pairs are, and then calculate how many such descriptions there are.

(b)

Knowlets:	RawKey, RawKeyKeyDiff ProblemID: RawKeyDiff
Time:	
Difficulty:	

In the lecture, we claimed that if $\rho \in S^{\text{test}}_{\text{Ideal}}$, and we measure A's and B's system in the computational basis, then with probability 1, we have $|K_A \oplus K_B| \leq t$.

Show that this is true.

Hint: If you have trouble, start small. First show it for a state $|\widetilde{xy}\rangle$ with $|xy| \leq t$. Then show it for a pure state $|\Psi\rangle$ that is a superposition of such $|\widetilde{xy}\rangle$ (like the ones that occur in the definition of $S^{\text{test}}_{\text{Ideal}}$. And then got for $\rho \in S^{\text{test}}_{\text{Ideal}}$.

2 Inverting cyclic functions

Consider a function $H : [N] \to [N]$ where $[N] := \{0, \ldots, N-1\}$. Let $H^i(x)$ denote $H(H(H(\ldots H(x)\cdots)))$ (applied i times). For the sake of this problem, we call H cyclic if there exists a value p (the period) such that for all x, $H^p(x) = x$.

(a)

Knowlets:	DFT, Shor ProblemID: HashPeriodAlgo
Time:	
Difficulty:	

Let $U_H|x\rangle|i\rangle|0\rangle = |x\rangle|i\rangle|H^i(x)\rangle$. Give a quantum algorithm involving U_H for finding the period of H (assuming that H is cyclic).

Note: You may assume that the DFT D_N can be implemented as a polynomial-time[1] quantum circuit. (This is, in general, not true for all N. But in the general case, you would be able to use an approximately solution that is only slightly more complicated than the solution needed here.)

Note: "involving U_H" means that you can apply U_H in a single runtime step.

(b)

Knowlets:	ProblemID: HashPeriodInvert
Time:	
Difficulty:	

(Bonus problem) Given $y = H(x)$ and given the period of p, show that you can find x in polynomial-time. (You may still use U_H.)

(c)

Knowlets:	ProblemID: HashPeriodWrong
Time:	
Difficulty:	

The following statement is wrong:

> Given a cyclic H and a value $y \in \operatorname{range} H$, using the algorithm from Problem 2 (a), we can find the period p of H, and then using the algorithm from Problem 2 (b), we can compute $H^{-1}(y)$.[2] Moreover, all involved algorithms run in polynomial-time. Hence using quantum computers, cyclic functions can be inverted in polynomial-time.

Why?

[1] By polynomial-time, I mean that the size of the circuit is bounded by $p(\log N)$ for some polynomial p.

[2] Notice that cyclicity implies bijectivity, so H^{-1} is well-defined.