CS 6846: Quantum Algorithms and Cryptography Oct 11, 2022

Homework 3

Instructor: Shweta Agrawal *Due: Oct 22, 5 pm.*

Instructions:

1. Please type up your solutions using latex.
2. You may collaborate with other students. Please mention the names of your collaborators or any other source that you use for the solution.
3. Please type up your solutions *individually* without any help.

Problem 1: Fun with Grover's Algorithm (2+3+3 pts).

In class we saw Grover's algorithm for the case where there was only one accepting input to the function. This can be generalized to the case where there are multiple accepting inputs. Let us assume that function $F : \{0,1\}^n \to \{0,1\}$, accepts a δ fraction of its inputs, i.e. $\Pr_x \left(f(x) = 1\right) = \delta$. Then we can find one accepting input with $O(\sqrt{1/\delta})$ evaluation queries to F.

We will consider the application of Grover's algorithm to collision finding. We are given that a function $G : \{0,1\}^m \to \{0,1\}^n$ where $m > n$, is 2 to 1: for each x_i there is exactly one other x_j such that $G(x_i) = G(x_j)$. Such an (x_i, x_j) pair is called a collision. Let $N = 2^n$.

1. Suppose S is a randomly chosen set of s elements in the domain of G. What is the probability that there is a collision in S?
2. Give a classical randomized algorithm that finds a collision (with probability $\geq 2/3$) using $O(\sqrt{N})$ queries to F.
   Hint: Use $1 - x \leq e^{-x}$ for $x \in [0,1]$. What is the above probability for $s = 2\sqrt{N}$?
3. Give a quantum algorithm that finds a collision (with probability $\geq 2/3$) using $O(N^{1/3})$ queries.
   Hint: Choose a set of size $s = N^{1/3}$ and classically query its elements. If this set does not contain a collision, use Grover's algorithm to find one. How will you define a function that is compatible with Grover's algorithm?

Problem 2: Generalizing Collisions (5 pts).

A collision can be thought of as follows: two distinct inputs x_0, x_1 such that $G(x_0) \oplus G(x_1) = 0^n$. Consider the following generalization: given $G : \{0,1\}^m \to \{0,1\}^n$ for $m >> n$, find 3 distinct inputs x_0, x_1, x_2 such that $G(x_0) \oplus G(x_1) \oplus G(x_2) = 0^n$. Explain how to solve this problem in time $O(N^{1/4})$ using Grover's algorithm.

Problem 3: Breaking Another PKE (6 pts).

Let $P : \{0,1\}^n \to \{0,1\}^n$ be a permutation; that is, a function without any collisions. Let $Q(x) = P(x \oplus k_0) \oplus k_1$ for some secret keys k_0, k_1. It is known that if you can only make classical queries to these two functions, then you cannot recover k_0, k_1. This fact is used in the design of encryption schemes: P is a public permutation that everyone knows, and you turn it into a private permutation

Q as above. Then Q can be used to encrypt messages (decryption will require the ability to compute the inverse of P, but we will ignore it for this problem).

Show that quantum queries to both P and Q allow for the recovery of k_0, k_1.
Hint: try defining a function f based on P and Q such that f is an instance of Simon's problem.

Problem 4: Rethinking Shor's Algorithm (5 pts).

In class, we saw that Shor's algorithm has the following broad steps: i) create a uniform superposition of inputs, ii) evaluate the function f_a on this superposition, iii) measure the registers containing the function value to get $f_a(x) = y$, iv) apply QFT on data registers to obtain a superposition of points that are separated by M/r (ignore the rounding issue for now), v) measure to obtain a multiple of M/r, vi) repeat to obtain many such multiples and recover r via GCD, since M is known.

Step (iii) gives a superposition of points that are separated by period r. In more detail, for some x_0, we obtained a state:

$$\frac{1}{\sqrt{M/r}} \sum_{j=0}^{M/r-1} |x_0 + jr, y>$$

Suppose we measure the data registers at this stage, i.e. *before* applying the QFT. The rationale is that we already have a superposition of points that are separated by r, so why can't we obtain many values of $x_0 + jr$, subtract pairs to remove the x_0 to obtain many multiples of r and then take GCD to recover r? Does this work? If so, why? If not, why?

Problem 5: Application of Grover's Algorithm (4 pts)

Let $N = 2^n$ and $x_0, ..., x_{N-1}$ be a sequence of distinct integers (you can think of them as the outputs in the truth table of some function F). We can query this function in the usual way, i.e., we can apply unitary $O : |i, 0\rangle \to |i, x_i\rangle$, as well as its inverse. The minimum of F is defined as $min\{x_i | i \in \{0, ..., N-1\}\}$. Give a quantum algorithm that finds (with probability $\geq 2/3$) an index achieving the minimum, using $O(\sqrt{N} \log N)$ queries.

Hint: start with $m = x_i$ for a random i, and repeatedly use Grover's algorithm to find an index j such that $x_j < m$ and update $m = x_j$. Continue this until you can find no element smaller than m, and analyze the number of queries of this algorithm. You are allowed to argue about this algorithm on a high level. Bonus: give a quantum algorithm that uses $O(\sqrt{N})$ queries.

Problem 6: Lattices (4+3 pts)

1. Prove that two bases $B_1, B_2 \in \mathbb{R}^{m \times n}$ generate the same lattice, i.e. $\mathcal{L}(B_1) = \mathcal{L}(B_2)$ if and only if $B_2 = B_1 U$ for some unimodular matrix U.

2. In class we saw the random lattices used in cryptography, as well as hard problems on lattices such as shortest vector and closest vector problem. Express the SIS and LWE problems that we saw in class as lattice problems.