Homework 4

Due: Tuesday October 27, 11:59pm; email the pdf to pgarriso@andrew.cmu.edu

Solve any 5 out of 7

1. **[Learning parities.]** Suppose $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ is a parity function, meaning that $f(x) = \sigma \cdot x$ for some (possibly unknown) $\sigma \in \mathbb{Z}_2^n$. In this problem, we will consider the query complexity of learning σ by algorithms with *zero-error*, meaning that they always output the correct answer.

   (a) How many queries are needed to classically learn σ with zero-error? Give an algorithm for this problem, and show that it is optimal.

   (b) How many queries are needed to quantumly learn σ with zero-error? Give an algorithm for this problem, and show that it is optimal.

   (c) Explain why even if we allow the classical algorithm to fail with some fixed probability (e.g., probability $\frac{1}{3}$), it requires the same asymptotic query complexity as the zero-error case.

2. **[A quantum algorithm for discrete Log.]** Let $M \geq 2$ be an integer, let $\mathbb{Z}_M^*$ denote the group of invertible integers mod M (i.e., those $a \in \mathbb{Z}_M$ with $\gcd(a, M) = 1$). In the discrete logarithm problem, we are given as input $M > 0$ and a "generator" g; i.e., an integer with $\{g^0, g^1, g^2, g^3, \ldots, g^{N-1}\} = \mathbb{Z}_M^*$, where $N = |\mathbb{Z}_M^*|$. We will assume that we "know" N; this assumption is removed in the next problem. We are further given as input a number $a \in \mathbb{Z}_M^*$, and the goal is to find its "logarithm" with respect to g, meaning the smallest number $\ell \in \mathbb{Z}_N$ such that $g^\ell \equiv a \pmod{M}$.

   (a) Define the function $f : \mathbb{Z}_N \times \mathbb{Z}_N \to \mathbb{Z}_M^*$ by $f(x, y) = a^x g^y \pmod{M}$. Show that f is an instance of the hidden subgroup problem on $\mathbb{Z}_N \times \mathbb{Z}_N$ (i.e. exhibit a subgroup H of $\mathbb{Z}_N \times \mathbb{Z}_N$ and show that f assigns unique colors to its cosets).

   (b) Because f is efficiently computable (by modular exponentiation), we can assume that we have access to an oracle of the form $O_f : |x\rangle |y\rangle |z\rangle \mapsto |x\rangle |y\rangle |z \oplus f(x, y)\rangle$, where $x, y \in \mathbb{Z}_N$ and $z \in \{0, 1\}^m$, where m is the number of bits used to represent elements of $\mathbb{Z}_M^*$. Using this oracle, give an efficient quantum algorithm for computing ℓ. You may assume that we can efficiently prepare the uniform superposition over $\mathbb{Z}_N$ and compute the Fourier transform over $\mathbb{Z}_N$ (i.e. the unitary map $|x\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{\gamma \in \mathbb{Z}_N} \chi_\gamma(x)^* |\gamma\rangle$) even if N is not necessarily a power of two.

3. **[Discrete log miscellany.]**

   (a) In a *public key exchange protocol*, two parties, Alice and Bob, would like to agree on a secret shared string (called the *key*) which they may use for a purpose such as cryptography. We imagine that they are spatially separated by a great distance, so their only means of communication is via a public channel which some eavesdropper Eve may be listening in on. Their goal is to use this channel to agree on a secret key which Eve is unable to guess. Consider the following protocol:

i. Alice and Bob publicly agree on a prime p and a generator g of $\mathbb{Z}_p^*$.

ii. Alice selects a private integer a and Bob selects a private integer b. (It is a fact that $\mathbb{Z}_p^*$ is a cyclic group of order $p-1$, so we may assume $a, b \in \mathbb{Z}_{p-1}$.)

iii. Alice sends Bob $g^a \pmod{p}$ and Bob sends Alice $g^b \pmod{p}$ on the public channel.

Show that using only the information publicly available and their respective private numbers, and without any further communication, Alice and Bob can agree on a private key which Eve is unlikely to guess from seeing only the information publicly available (unless Eve has access to an efficient algorithm for the discrete logarithm). The discrete logarithm also appears in other cryptographic protocols, e.g. the ElGamal public key cryptosystem.

(b) Given a number M, define the function $\phi(M) := |\mathbb{Z}_M^*|$. If $M = p_1^{k_1} \cdots p_f^{k_f}$, where $p_1, \ldots, p_f$ are distinct primes, then a well-known formula states that

$$\phi(M) = p_1^{k_1-1}(p_1-1) \cdots p_f^{k_f-1}(p_f-1).$$

(You might find it edifying to prove this fact in your free time.) In Problem 2, we assumed that we knew the order of $\mathbb{Z}_M^*$. Show that this assumption can be efficiently removed on a quantum computer.

(c) Show that if $M = pq$ for two distinct primes p and q, then we can factor M efficiently (with a classical computer) if we know $\phi(M)$. In general, it is known that the problems of factoring and computing the $\phi(\cdot)$ function are equivalent, meaning that they efficiently reduce to each other.

4. **[Solving hidden-shift by reduction to HSP for the dihedral group.]** The *dihedral group* D_N is defined as the "group of symmetries of the regular N-vertex polygon"; in other words, it is the automorphism group $\mathrm{Aut}(C_N)$ of the N-vertex cycle graph C_N.

(a) Show that there are two elements $x, y \in D_N$ satisfying $x^N = 1$ and $y^2 = 1$ and $xyxy = 1$. Furthermore, show that every element in D_N is of the form $y^a x^b$, where $a \in \mathbb{Z}_2$ and $b \in \mathbb{Z}_N$. Finally, show that $|D_N| = 2N$.

(b) Show that the HSP on D_N is easy quantumly in the case when the hidden subgroup H is generated by an element of the form x^b.

(c) Recall the *hidden-shift* problem from Lecture 5: Here there are two functions $f, g : \mathbb{Z}_N \to$ "colors" which output N distinct colors. Furthermore, we have the guarantee that $f(x) = g(x+s)$ for all $x \in \mathbb{Z}_N$, where $s \in \mathbb{Z}_N$ is an unknown hidden shift. Show that finding s reduces to the HSP on D_N.

5. **[Even more extensions to Grover.]** In this problem you may cite the results Homework 3, Problem 3, including the Bonus part. Suppose we are given quantum query access O_w to a binary string $w \in \{0,1\}^N$ (you may assume N is a power of 2, as usual).

(a) Show that there is a quantum query algorithm that: a) outputs $w \in \{0,1\}^N$ with probability 1; b) in expectation, makes $O(\sqrt{kN})$ queries to O_w, where k is the Hamming weight (number of 1's) in w.

(b) For an integer $0 \le k \le N$, let $T_k : \{0,1\}^N \to \{0,1\}$ be the function that has value 1 on input w iff the Hamming weight of w is at least k. Show that there is a quantum query algorithm computing $T_k(w)$ with high probability, making $O(\sqrt{k(N-k+1)})$ queries to O_f.

6. **[Symmetrization.]** Suppose $Q(w_1, \ldots, w_N)$ is a *symmetric* multilinear polynomial of degree at most d; here *symmetric* means that Q is unchanged under any permutation of the variables. Prove that there exists a *univariate* polynomial q of degree at most d such that $Q(w_1, \ldots, w_N) = q(z)$ for all $w = (w_1, \ldots, w_N) \in \{0,1\}^N$, where z denotes $w_1 + \cdots + w_N$.

7. **[Classical vs. quantum decision tree complexity.]** Let $F : \{0,1\}^N \to \{0,1\}$. Imagine there is an unknown $w \in \{0,1\}^N$ to which we have query access (either quantum or classical). We wish to compute $F(W)$. Let $D(F)$ denote the least number of queries needed[1] for a classical deterministic algorithm, and let $Q(F)$ denote the least number of queries needed for a quantum algorithm[2].

   Define the *embedded-OR complexity* of F, denoted $\mathrm{eo}(F)$, as follows: First, given a string $x \in \{0,1\}^N$ and a subset $S \subseteq [N] \coloneqq \{1, 2, \ldots, N\}$, let $x^{\oplus S}$ denote the string x with all its bits in the S positions negated. Second, if $S_1, \ldots, S_t$ are disjoint subsets of $[N]$, say that x *is flippable on* $S_1, \ldots, S_t$ if $F(x) \neq F(x^{\oplus S_j})$ for all $1 \leq j \leq t$. Finally, $\mathrm{eo}(F)$ is defined to be the largest t such that there exists $x \in \{0,1\}^N$ and disjoint $S_1, \ldots, S_t \subseteq [N]$ such that x is flippable on $S_1, \ldots, S_t$.

   (a) Show that $Q(F) \geq \Omega(\sqrt{\mathrm{eo}(F)})$.

   Define the *certificate complexity* of F, denoted $C(F)$, as follows: First, given $x \in \{0,1\}^N$, we define $C_x(F)$ to be the size of the smallest set $S \subseteq [N]$ such that $F(y) = F(x)$ for all $y \in \{0,1\}^N$ such that y and x agree on the positions S. The values of x in the positions S are called a "certificate" for $F(x)$. Finally, define $C(F) = \max_x C_x(F)$. Another way to look at it is that $C(F)$ is the least number of queries a "psychic" algorithm (knowing w) would have to make in order to "certify" to a bystander what the value of $F(w)$ is.

   (b) Show that $C(F) \leq \mathrm{eo}(F)^2$. (Hint: show that if x is flippable on $S_1, \ldots, S_t$ and the set S_j is minimal, then $|S_j| \leq \mathrm{eo}(F)$.

   (c) Show that $D(F) \leq C(F)\mathrm{eo}(F)$, and thereby deduce that $D(F) \leq O(Q(F)^6)$. (Hint: consider the following strategy. At any given time, pick a "certificate", consistent with the query results so far, that would force F to be 1 if w were consistent with it. Query all the positions in that certificate to see if w is indeed consistent with it. Show that one has to repeat this at most $\mathrm{eo}(F)$ times.)

[1] For the best algorithm on its worst w.

[2] For the best algorithm that succeeds on all inputs with probability at least 2/3.