CMPT 476/776: Introduction to Quantum Algorithms
Assignment 4

Due **March 26th, 2026 at 11:59pm on crowdmark**
Complete individually and submit in PDF format.

Question 1 [7 points]: Coset states and Generalized Simon

Recall that the dot product on the vector space $\mathbb{Z}_2^n$ is defined as $x \cdot y = x_1 y_1 \oplus x_2 y_2 \oplus \cdots \oplus x_n y_n$ where $x = (x_1, x_2, \ldots, x_n), y = (y_1, y_2, \ldots, y_n) \in \mathbb{Z}_2^n$. For any subspace S of $\mathbb{Z}_2^n$, define the orthogonal complement of S with respect to the dot product as

$$S^\perp = \{z \in \mathbb{Z}_2^n \mid s \cdot z = 0 \quad \forall s \in S\}.$$

1. Let $|x + S\rangle = \frac{1}{\sqrt{|S|}} \sum_{s \in S} |x + s\rangle$ and show that

$$H^{\otimes n}|x + S\rangle = \sqrt{\frac{|S|}{2^n}} \sum_{z \in S^\perp} (-1)^{x \cdot z} |z\rangle$$

   Hint: show that for any $z \in \mathbb{Z}_2^n$, either $z \in S^\perp$ (i.e. $z \cdot s = 0$ for all $s \in S$) or $z \cdot s = 0$ for exactly **half** the elements $s \in S$.

2. Show that Simon's algorithm can be generalized to solve the *Boolean hidden subgroup problem* **with no changes to the quantum part**. That is, given $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$ such that $f(x) = f(y)$ if and only if $x = y \oplus s$ for some s in a **non-trivial** linear subspace S of $\mathbb{Z}_2^n$, generalize Simon's algorithm to find a **basis** for S. You should sketch an algorithm in pseudo-code.

Question 2 [3 points]: Factoring, classically

In this question we will factor the number 21 classically. You do not have to show your calculations and you may find it useful to use a calculator or program to calculate the GCD. If it were me, I would probably write a program to do it.

1. Compute the period of $f(x) = 5^x \mod 21$ — that is, find the smallest integer r such that $5^r \equiv 1 \mod 21$.

2. Compute $GCD(5^{r/2} + 1, 21)$, $GCD(5^{r/2} - 1, 21)$. What's the problem?

3. Now repeat steps 1 and 2 with $f(x) = 2^x \mod 21$ to factor 21 into its prime factors.

Question 3 [4 points]: QFT or QFT^{-1}?

In class we've been pretty cavalier about whether we use QFT or the $QFT^{-1} = QFT^\dagger$ in period finding and phase estimation. In this question we'll investigate why.

1. Determine what transformation is applied by $QFT_{2^n}^2$ — that is, compute $QFT_{2^n}(QFT_{2^n}|x\rangle)$ where $x \in \{0,1\}^n$.
2. Use the result of the previous question to determine the **order** of the QFT (i.e. the minimal $k \in \mathbb{Z}$ such that $QFT^k = I$).
3. Now suppose you accidentally applied QFT when you should have applied QFT^{-1} and measured the result to get a bit string $y \in \{0,1\}^n$. How could you **classically** recover from y the "correct" bit string $z \in \{0,1\}^n$ which you would have measured if you had instead applied QFT^{-1}?

Question 4 [5 points]: Modular multiplication

Recall that Shor's algorithm requires an (expensive) oracle for performing the classical function

$$x \mapsto a^x \mod M,$$

where $a^x \mod M$ can be written as a sequence of multiplications $a^{x_0}(a^2)^{x_1}\cdots(a^{2^{n-1}})^{x_{n-1}} \bmod M$.

1. Let $M = 2^n$. Give **classical** pseudo-code for adding two length n-binary integers mod 2^n.
2. Using addition mod 2^n as a sub-routine, give pseudo-code for binary multiplication mod 2^n.
3. Give an expression in big-O notation for the complexity of modular exponentiation mod 2^n using the sub-routines you developed in parts 1 and 2.

Question 5 [7 points]: Qutrit quantum computing

Much of quantum computation can be generalized to higher-dimensional **qudits**. Most gates we've seen have higher-dimensional generalizations, like the **Pauli gates** X, Y, Z and the Hadamard or **Fourier** gate H. In this question we will explore this notion briefly.

Consider a **qutrit**, which is a 3-dimensional quantum state — i.e. a unit vector in $\mathbb{C}^3$. As discussed in class, we denote the computational basis of $\mathbb{C}^3$ as $\{|0\rangle, |1\rangle, |2\rangle\}$, or $|x\rangle$ where $x \in \mathbb{Z}_3$, the integers mod 3. Denote the primitive third root of unity as $\omega_3 = e^{2\pi i/3}$. The Pauli X and Z operators on a qutrit can now be defined as

$$X = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad Z = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \omega_3 & 0 \\ 0 & 0 & \omega_3^2 \end{bmatrix}$$

Likewise, the qutrit Hadamard gate can be defined as

$$H = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 1 & 1 \\ 1 & \omega_3 & \omega_3^2 \\ 1 & \omega_3^2 & \omega_3 \end{bmatrix}$$

1. Show that X and Z have order 3 (i.e. $X^3 = Z^3 = I$)

2. Show that $XZ = \omega_3^2 ZX$. Use this to calculate k (as a function of i and j) such that $X^i Z^j = \omega^k Z^j X^i$ for $i, j \in \{0, 1, 2\}$.

3. Show that $H^\dagger Z H = X$.

4. Compute the eigenvalues of X and give corresponding (unit) eigenvectors. Hint: recall the relationship between H and the eigenvectors of X in the qubit case.

5. Now show that Deutsch's algorithm generalizes to *qutrits*. Explicitly, given a function $f : \mathbb{Z}_3 \to \mathbb{Z}_3$ promised to either be **constant** or **balanced** where balanced in this case means for every $y \in \mathbb{Z}_3$, there exists exactly one $x \in \mathbb{Z}_3$ such that $f(x) = y$, show that Deutsch's algorithm with the qutrit version of the H gate works the same way.

   Hint: note that the qutrit H gate is the 3-dimensional QFT. That is,

$$H|x\rangle = \frac{1}{\sqrt{3}} \sum_{z \in \mathbb{Z}_3} \omega_3^{xz} |z\rangle.$$