North Carolina State University – Departments of Computer Science/ECE

CSC 591/ECE 592 Quantum Computing Fall 2019

Last Name: ______________________ First Name: ______________________

Sample Solutions – Midterm Exam – 10/8/19

Please read these instructions carefully before answering the questions:

- The exam is closed everything (books, notes). You may only use non-programmable calculators. You have 75 minutes.
- Write your answers in the provided space. If you run out of space, use an extra sheet with **your name on it**.
- Your work must be individual. Cheating will be punished instantly. Please focus on your own exam.
- The only person you can ask questions is the proctor / instructor. He/she will only give an answer if you detect an ambiguity in a question. Claims such as "I do not understand XYZ" will not be answered.

1. (10 points, 5 each) (a) What is the difference between adiabatic quantum computing (AQC) and adiabatic quantum optimization (AQO)? (b) How does their compute capability compare to gate-based quantum computing?

   (a) AQC is a more powerful than AQO as the latter's Hamiltontian has only one degree of freedom (Z) while the former has two (Z,X). Both of them have longitudinal interactions and fields for Z, but while AQC has transverse interactions and fields for X, AQO only features a traverse field for X with a single, global amplitude to control the annealing process.

   AQO: $\mathcal{H}(t) = -\sum_{i=0}^{N-2}\sum_{j=0}^{N-1} J_{i,j}\sigma_i^Z\sigma_j^Z - \sum_{i=0}^{N-1} S_i\sigma_i^Z - \Gamma(t)\sum_{i=0}^{N-1}\sigma_i^X$

   AQC: $\mathcal{H}(t) = -\sum_{i=0}^{N-2}\sum_{j=0}^{N-1} J_{i,j}\sigma_i^Z\sigma_j^Z - \sum_{i=0}^{N-1} S_i\sigma_i^Z - \sum_{i=0}^{N-2}\sum_{j=0}^{N-1} K_{i,j}\sigma_i^X\sigma_j^X - \sum_{i=0}^{N-1}\Delta_i\sigma_i^X$

   Notice: All terms are negative for minimization. But all terms positive is also a correct description of the general Hamiltonian (without bias). The Hamiltonians do not need to be spelled out but their terms should be described in the solution.

   (b) AQO is constrained to only optimization problems while AQC is more general. In fact, AQC is equivalent to gate programming with a universal set of gates. (This argument is still theoretical and does not consider noise.)

2. (10 points) In a homework, you modified a factoring code for the D-Wave. What are the differences between D-Wave's factoring and Shor's factoring algorithm? Discuss.

   Shor's algorithm lowers computational complexity from exponential to logarithmic (raised to a constant), which is certainly showing quantum supremacy. The idea is to use a quantum Fourier transform to find the period of a sequence some (systematically tried) number that does not divide the two prime factors in question. Once a sequence is found, it is classically checked and confirmed or refuted, and in the latter case another sequence is inquired from the quantum side.

   D-Wave's factoring is brute force and does NOT result in any reduction of computational complexity. It enumerates all sampled solutions obtained by annealing to provide a pair of factors for a large number. Correct answers are indicated by minimal energy but need to be checked classically in practive since noise may purturb the answers. However, it is the brute-force enumeration that does not provide supremacy here. To date, not even quantum advantage has been demonstrated by D-Wave for this problem since heuristic (classical) approaches are at least at par with D-Wave's solution to the probem.

3. (10 points) Derive the 3-qubit Hamiltonian / Qubo from an equation for the following problem any way you like (e.g., using the table below), show the resulting weights and strengths and verify that the objective function is larger than ground for any qubit permutation that is *not* a valid solution while being ground of the valid solutions.

   $\neg a \wedge b = c$ for $q_i^z \in 0, 1$:

a	*b*	*c*	*Obj*
0	*0*	*0*	*T =0*
0	*0*	*1*	*>T =1*
0	*1*	*0*	*>T =1*
0	*1*	*1*	*T =0*
1	*0*	*0*	*T =0*
1	*0*	*1*	*>T =3*
1	*1*	*0*	*T =0*
1	*1*	*1*	*>T =1*

   $Obj = b + c - ab + 2ac - 2bc = 0$

   So weights/strengths are $(0, 1, 1, -1, 2, -2)$

   Obj(a,b,c) as indicated in table

4. (10 points) Embeddings:

   Given the 3-qubit Hamiltonians for $a \vee b = c$ (OR relation) as $H = a+b+c+ab-2ac-2bc$ and $a \wedge b = c$ (AND relation) as $H = 3c + ab - 2ac - 2bc$, and $\neg a = b$ (NOT relation) as $H = -a - b + 2ab$, create an imply relation by composing the given relations as follows: $a \implies b = c \equiv \neg a \vee a \wedge b = c$, i.e., using AND/OR/NOT. Draw the resulting embedded graph with weights and strengths and indicate its Hamiltonian / Qubo.

   Let $d = \neg a, e = a \wedge b$.

```
a(-1)..1..d(0).
 |  .\     |   \-2
 1    -2   1    .c(1)
 |      \. |   /-2
b(0)..-2..e(4).
```

   $H = -a + c + 4e + ab + 2ad - 2ae - 2be - 2cd - 2ce + de$

5. (10 points) Show by calculation whether the state $\frac{1}{\sqrt{2}}(|0\rangle|+\rangle + |1\rangle|-\rangle)$ is entangled.

   Solution:

$$\frac{1}{\sqrt{2}}(|0\rangle|+\rangle + |1\rangle|-\rangle) = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle - |11\rangle)$$

   If the state is not entangled, then there must be some values a_1, b_1, a_2, and b_2, such that:

$$(a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle) = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle - |11\rangle)$$

   This requires that all of the following must be true:

$$\begin{aligned} a_1 \cdot a_2 &= \tfrac{1}{2} \\ a_1 \cdot b_2 &= \tfrac{1}{2} \\ a_2 \cdot b_1 &= \tfrac{1}{2} \\ b_1 \cdot b_2 &= -\tfrac{1}{2} \end{aligned}$$

   If a solution exists, b_1 and b_2 must have opposite signs. The second and third equations require that a_1 has the same sign as b_2 and that a_2 has the same sign as b_1. This implies that a_1 and a_2 must also have opposite signs. However, from the first equation, a_1 and a_2 must have the same sign. Therefore, the system has no solution and the state is entangled.

6. (5 points) Given the state $|+\rangle$ show that this state can be entangled, or provide an explanation why it cannot be entangled.

   Solution:

   $|+\rangle$ is not entangled. It represents a single qubit. Entanglement must involve two qubits.

7. (10 points) TRUE / FALSE: Mark whether each statement is true (T) or false (F).

   (a) _____ To obtain an expectation value of an observable, one must add all of the probability amplitudes to get the final measured result.

   (b) _____ The probability amplitude has an indeterminate specific value until a measurement is performed.

   (c) _____ For a quantum mechanical system, once a measurement is done on the system, all information prior to that measurement is permanently lost.

   (d) _____ Step by step evolution from an initial state on a quantum computer to some final state must maintain the initial probability amplitude through to the final measurement.

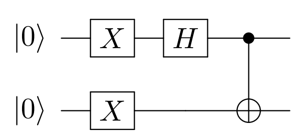
   Solution:

   (a) False, (b) True, (c) True, (d) False

8. (15 points) We have several times shown the circuit that generates a Bell state: $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. There are actually four Bell states, shown below:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$
$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$$
$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$$
$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$$

   (a) (5 pts) Draw a quantum circuit that generates the $|\Psi^-\rangle$ state, starting from an initial state of $|00\rangle$.
   Solution:



(b) (5 pts) Write qiskit code to create the circuit, and measure the output. Don't include code to simulate the circuit; just create it.

Solution:

```
q = QuantumRegister(2)
c = ClassicalRegister(2)
qc = QuantumCircuit(q, c)
qc.x(q)
qc.h(q[1])   # circuit is drawn differently than above, but output matches
qc.cx(q[1],q[0])
qc.measure(q,c)
```

(c) (2.5 pts) Using qiskit's `qasm_simulator` backend, can we distinguish between states $|\Psi^+\rangle$ and $|\Psi^-\rangle$? If so, how? If not, why not?

Solution:

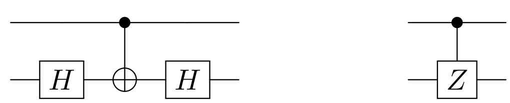
No. Measuring the output will square the amplitude, the probability of seeing output `10` is 0.5 in both cases.

(d) (2.5 pts) Using qiskit's `statevector_simulator` backend, simulating *exactly* the same circuit, can we distinguish between states $|\Psi^+\rangle$ and $|\Psi^-\rangle$? If so, how? If not, why not?

Solution:

Yes, but it may require multiple simulations. Measuring the output will force the state into either `01` or `10` with equal probability. So we will see an amplitude of 1 `01` and zero everywhere else, or we will see an amplitude of -1 for `10` and zero everywhere else.

9. (10 points) Prove that the two circuits below are equivalent.



As a reminder: $X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, $Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$, and $H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

Solution:

$$
\begin{aligned}
(I \otimes H)(CNOT)(I \otimes H) &= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \\
&= \frac{1}{2} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \\
&= \frac{1}{2} \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & -2 \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} = CZ
\end{aligned}
$$

10. (10 points) Draw a reversible quantum circuit that "detects" an input pattern of $|1100\rangle$. In other words, the output qubit is flipped whenever the other four input qubits match the pattern. You may use Toffoli, CNOT, and any single-qubit gates. Label your qubits to disambiguate between the most-significant and least-significant.

    If ancilla qubits are used, they must be initialized to $|0\rangle$ and restored to $|0\rangle$.

    Solution: