Please write your name in the upper corner of each page.

The exam is open book, open notes.

Questions vary substantially in difficulty. Use your time wisely.

Partial Credit: For all of the problems on the exam except the True/False problems, if you explain your work you may receive partial credit for an incorrect answer.

Feel free to use extra paper for your solutions. Please put your name on the sheet and clearly label which parts are on it. Include only one problem per sheet.

<table>
<thead>
<tr>
<th>Problem</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>200</td>
<td></td>
</tr>
</tbody>
</table>
**Problem 1: True-False** [32 points: 4 points per question / 1 point per item]
Mark each item as True or False by putting a T or F in each box. Write clearly; ambiguous or hybrid T-F’s won’t get credit.

1. Which of the following languages are decidable?
   - □ CLIQUE
   - □ \{\langle M \rangle : M is a Turing machine and L(M) = CLIQUE\}
   - □ \{\langle M \rangle : M is a Turing machine that recognizes an NP-complete language\}
   - □ \{\langle M \rangle : M is a Turing machine that recognizes a language that is also recognized by some other Turing machine M’ with an even number of states\}.

2. Which of the following statements are true about the Post Correspondence Problem (PCP)?
   - □ \text{A}_{TM}, the acceptance problem for ordinary Turing machines, is mapping-reducible to \text{PCP}.
   - □ \text{PCP} is mapping reducible to \text{A}_{DFA}, the acceptance problem for DFAs.
   - □ \text{INF} is mapping reducible to \text{PCP}, where \text{INF} is the set of Turing machines that accept infinitely many strings.
   - □ \text{PCP} is mapping reducible to \text{INF}.

3. Which of the following statements are true about probabilistic computing?
   - □ If a polynomial-time randomized machine can decide a problem with error probability exactly 0, then that problem is in P.
   - □ If a decision problem is in BPP, then there exists a probabilistic Turing Machine that decides it with two-sided error probability at most \frac{1}{2^n}, where n is the length of the input.
   - □ If there exists a Turing Machine that correctly decides a problem for at least 90% of inputs of length n for all n, then that problem is in BPP.
   - □ A BPP machine with a BPP oracle can solve problems outside of BPP.

4. Which of the following are known to be true statements about branching program equivalence testing?
   Here, p is a large prime number. Assume B_1 and B_2 are over the same set of Boolean variables.
   - □ \text{EQ}_{BP}, the equivalence problem for branching programs, is in P.
   - □ If B_1 and B_2 are equivalent branching programs over the same set of Boolean variables, then they always produce the same output when evaluated with values in \mathbb{Z}_p, the integers mod p.
   - □ If B_1 and B_2 are inequivalent branching programs, then they produce different outputs for at least half of the possible assignments of Boolean values to the variables.
   - □ If B_1 and B_2 are inequivalent branching programs, then they produce different outputs for at least half of the possible assignments of values in \mathbb{Z}_p to the variables.
5. Which of the following statements are known to be true about cryptography?

- If $P \neq NP$, then there exists a one-way function.
- If there exists a one-way function, then there exists a pseudorandom generator.
- If there exists a pseudorandom generator, then there exists a secure public-key cryptosystem.
- If there exists a pseudorandom generator, then there exists an NP problem that is hard on average.

6. Which of the following statements are known to be true about PAC-learning?

- For any polynomial $p$, the concept class of all functions computable by Boolean circuits of size $p(n)$ is PAC-learnable in polynomial time.
- If all PAC-learning problems are solvable in polynomial time, then RSA is insecure.
- There exists a concept class with finite VC-dimension that requires infinitely many samples to PAC-learn.
- There exists a concept class with an infinite sample space that requires finitely many samples to PAC-learn.

7. Which of the following statements are true about quantum information?

- The states $\frac{|0\rangle + |1\rangle}{\sqrt{2}}$ and $\frac{|0\rangle - |1\rangle}{\sqrt{2}}$ can be distinguished perfectly.
- The states $\frac{|0\rangle + |1\rangle}{\sqrt{2}}$ and $|0\rangle$ can be distinguished perfectly.
- The Bell Inequality implies that Alice and Bob can exploit quantum entanglement to transmit messages faster than light.
- Every unitary transformation is reversible.

8. Which of the following statements are known to be true about quantum computing?

- Quantum computers can solve NP-complete problems in polynomial time.
- Quantum computers can factor integers exponentially faster than classical computers can do so.
- In the black-box model, there exist problems that quantum computers can solve exponentially faster than classical computers.
- One can approximate any $n$-qubit unitary operation arbitrarily well using $O(n^c)$ Hadamard and Toffoli (CCNOT) gates, for some constant $c$. 
Problem 2: (20 points: 5, 15)
If $x = a_1a_2 \ldots a_n$ and $y = b_1b_2 \ldots b_n$ are two strings of the same length $n$, define $\text{alt}(x, y)$ to be the string in which the symbols of $x$ and $y$ alternate, starting with the first symbol of $x$, that is, $a_1b_1a_2b_2 \ldots a_nb_n$. If $L$ and $M$ are languages, define $\text{alt}(L, M)$ to be the language of all strings of the form $\text{alt}(x, y)$, where $x$ is any string in $L$ and $y$ is any string in $M$ of the same length.

(a) [5 points] Write a regular expression for $\text{alt}(0^*1, 01^*)$, where the regular expressions represent languages.

(b) [15 points] If $L$ and $M$ are regular, must $\text{alt}(L, M)$ be regular? Either give a construction that proves this, or demonstrate a counterexample.
**Problem 3:** (20 points: 7, 3, 10, [10ec])

For each \(n\), let \(p(n)\) be the fraction of strings in \(\{0, 1\}^n\) that represent a zero-input Turing machine that halts when run starting from a blank tape. This means that a \(1 - p(n)\) fraction of these strings are either invalid Turing machine representations, or represent machines that run forever.

(a) [7 points] Describe an algorithm \(M(i, n)\) to compute asymptotically good lower bounds \(p_i(n)\) that converge to \(p(n)\) from below as \(i \to \infty\). This means that \(p_i(n) \leq p(n)\), and that for every \(n\) there are only finitely many \(i\) for which \(p_i(n) \neq p(n)\).

(b) [3 points] Why doesn’t the algorithm in part (a) immediately let you compute \(p(n)\)?

(c) [10 points] Show that the function \(p(n)\) is not computable.

(d) [Extra credit, 10 points] Show that there is no an algorithm \(M'(i, n)\) that is like \(M(i, n)\) from (a), but computes upper bounds rather than lower bounds.
Problem 4: (25 points: 5, 5, 10, 5)
For any $k$, the language $k$-COLOR is defined to be the set of (undirected, not necessarily planar) graphs whose vertices can be colored with $k$ or fewer distinct colors, in such a way that no two adjacent vertices are colored with the same color. The language 3-COLOR is known to be NP-complete.

(a) [5 points] Prove that 2-COLOR is in P.

(b) [5 points] Prove that 4-COLOR is in NP.

(c) [10 points] Prove that 4-COLOR is NP-complete.

(d) [5 points] Assuming $P \neq NP$, for exactly which values of $k \in \{1, 2, 3, \ldots\}$ is $k$-COLOR NP-complete? Explain why.
Problem 5: (23 points: 5, 10, 8)

Define the probabilistic complexity class PP as follows: A language \( L \) is in PP if and only if there exists a probabilistic polynomial time Turing machine \( M \) such that:

- If \( w \in L \), then \( \Pr[M \text{ accepts } w] > \frac{1}{2} \).
- If \( w \notin L \), then \( \Pr[M \text{ accepts } w] \leq \frac{1}{2} \).

Show that:

(a) [5 points] \( \text{BPP} \subseteq \text{PP} \).

(b) [10 points] \( \text{NP} \subseteq \text{PP} \).

(c) [8 points] \( \text{PP} \subseteq \text{PSPACE} \).
Problem 6: (15 points: 10, 5, [10ec])
Recall the Diffie-Hellman key-exchange protocol from class: Alice and Bob publicly agree on a large prime number \( p \) and an integer \( 2 \leq g \leq p-1 \). Then, Alice generates a secret integer \( a \) and sends Bob \( g^a \pmod{p} \), while Bob generates secret integer \( b \) and sends Alice \( g^b \pmod{p} \). Finally, they send \( g^a \pmod{p} \) and \( g^b \pmod{p} \) to each other and Alice computes
\[
(g^b)^a = g^{ab} \pmod{p}
\]
and Bob computes
\[
(g^a)^b = g^{ab} \pmod{p},
\]
thereby establishing their shared secret key \( g^{ab} \).

(a) [10 points] Generalize the Diffie-Hellman protocol to a protocol in which three parties agree on a shared secret key.

(b) [5 points] What values does Eve know after she eavesdrops on a run of the three-party Diffie-Hellman protocol? Don’t forget the values she knows because they are public.

(c) [Extra credit, 10 points] Show that if the three-party Diffie-Hellman is secure, then so is regular two-party Diffie-Hellman.
Problem 7: (20 points: 5, 10, 5)
The following questions concern $n$-dimensional linear classifiers: that is, subsets of $\mathbb{R}^n$ that have the form
$$\{(x_1, \ldots, x_n) : a_1 x_1 + \ldots + a_n x_n \geq b\}$$
for some real numbers $a_1, \ldots, a_n$ and $b$.

(a) [5 points] Given as inputs two set $X$ and $Y$ of polynomially-many points in $\mathbb{R}^n$ (say, specified in binary notation), show that there’s a polynomial-time algorithm to decide whether there’s a linear classifier containing all points in $X$ and no points in $Y$.
(Note: You can use the fact that Linear Feasibility, the problem of deciding whether a given set of linear inequalities in $n$ real variables has a solution or not, is in P.)

(b) [10 points] Let $C$ be the concept class consisting of all linear classifiers. Show that $\text{VCdim}(C) \geq n+1$.

(c) [5 points] Can you conclude from part (b) that linear classifiers are PAC-learnable with finitely many samples? Explain why or why not.
Problem 8: (35 points: 5, 5, 5, 5, 5, 5, 5)

In this problem, we’ll show how Alice can “teleport” an arbitrary qubit $|\psi\rangle$ to Bob, despite sending only classical information.

Alice and Bob share in advance an entangled Bell state

$$\text{Bell}_{AB} = \frac{|00\rangle + |11\rangle}{\sqrt{2}}.$$  

Alice stays on Earth and holds qubit $A$, while Bob goes to the Moon and holds qubit $B$. Alice is given another qubit $X$ then she wants to send to Bob, whose state we’ll write as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. The goal of the protocol is to end with $|\psi\rangle$ in Bob’s qubit $B$. See the blackboard for a quantum circuit diagram.

(a) [5 points] Alice begins by applying a CNOT gate from her qubit $X$ to her qubit $A$. Write down the state of the 3-qubit system after this. Use the order $XAB$.

(Hint: Don’t use the $8 \times 8$ unitary matrix for CNOT acting on 3 qubits. Expand into basis states, and think about what the CNOT gate does to each basis state.)

(b) [5 points] Next, Alice applies a Hadamard gate to her qubit $X$. Write down the 3-qubit state that results.

(c) [5 points] Alice then measures both her qubits $X$ and $A$ in the standard basis. What is the probability that she gets the measurement outcome $|11\rangle$? If she does, what 1-qubit state is Bob left holding in $B$?
(d) [5 points] Bob’s state from part (c) is almost like the state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ that Alice wanted to send, but not quite. What $2 \times 2$ unitary operation can Bob apply to convert his state into $|\psi\rangle$?

(e) [5 points] The operations Bob needs to perform to get $|\psi\rangle$ depend on the outcome of Alice’s measurement. So, Alice must communicate to Bob which measurement outcome she got for Bob to know how to recover $|\psi\rangle$. Explain why any protocol to teleport a qubit requires communication from Alice to Bob.

(f) [5 points] The state $|\psi\rangle$ could involve unlimited information—for example, Alice could put entire text of Hamlet as the binary expansion of $\alpha$! Explain why the teleportation protocol doesn’t let Bob actually recover that much information, even if Alice and Bob can both perform unitary transformations with infinite precision.

(g) [5 points] It might seem that Alice has sent a copy of $|\psi\rangle$ to Bob, in violation of the No-Cloning Theorem. Explain why this is not the case.
Problem 9: (10 points: 5, 5, [10ec])
As mentioned in class, in 1996 Lov Grover discovered a quantum algorithm that searches a list of \( n \) items using only \( O(\sqrt{n}) \) steps. In this problem, we’ll see how Grover’s algorithm searches a four-item list in one step.

Let \( f \) be a Boolean function from two bits \{00, 01, 10, 11\} to \{0, 1\}. Throughout the problem, assume that there is exactly one input \( x \in \{00, 01, 10, 11\} \) for which \( f(x) = 1 \). We call this \( x \) the “marked item”; our goal is to find it with as few queries to \( f \) as possible.

(a) [5 points] In the worst case, how many queries to \( f \) does a classical algorithm need to make, before it has determined the location of \( x \)?

Now, consider the following quantum algorithm:

**Grover’s Algorithm for 4 Items:**

1. Begin with an equal superposition over 2-qubit states \(|00\rangle, |01\rangle, |10\rangle, |11\rangle\).
2. Multiply each basis state \(|x\rangle\) by \((-1)^{f(x)}\). In other words, negate the marked basis state.
3. Apply the unitary matrix

   \[
   U = \frac{1}{2} \begin{pmatrix}
   1 & -1 & -1 & -1 \\
   -1 & 1 & -1 & -1 \\
   -1 & -1 & 1 & -1 \\
   -1 & -1 & -1 & 1
   \end{pmatrix}
   \]

4. Measure both qubits and output the result.

(The unitary matrix in step 3 can be implemented by Hadamarding both qubits, negating \(|00\rangle\), then Hadamarding both qubits again.)

(b) [5 points] What is the probability that the above algorithm correctly returns the marked item?

(c) [Extra credit, 10 points] One can modify the above algorithm to search a list of eight items, by using three bits to represent each item, and replacing \( U \) by some \( 8 \times 8 \) unitary matrix \( V \). Explain why, regardless of what \( V \) you choose, such an algorithm cannot return the marked item with probability as good as your answer to (b).