Instructions: Turn in your answers to the questions below in class on or before the due date. You must include at least a brief summary of each question below as part of your answer (including the question number). Be sure your answers are legible. Note that each question is worth 1 point (for a total of 10 points).

1. Consider the following TM transition table for the language of strings that consist of zero or more a’s followed by zero or more b’s.

<table>
<thead>
<tr>
<th>Current State</th>
<th>Current Symbol</th>
<th>New State</th>
<th>New Symbol</th>
<th>Head Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>q₀</td>
<td>⊔</td>
<td>q_accept</td>
<td>⊔</td>
<td>R</td>
</tr>
<tr>
<td>q₀</td>
<td>a</td>
<td>q₁</td>
<td>a</td>
<td>R</td>
</tr>
<tr>
<td>q₀</td>
<td>b</td>
<td>q₂</td>
<td>b</td>
<td>R</td>
</tr>
<tr>
<td>q₁</td>
<td>⊔</td>
<td>q_accept</td>
<td>⊔</td>
<td>L</td>
</tr>
<tr>
<td>q₁</td>
<td>a</td>
<td>q₁</td>
<td>a</td>
<td>R</td>
</tr>
<tr>
<td>q₁</td>
<td>b</td>
<td>q₂</td>
<td>b</td>
<td>R</td>
</tr>
<tr>
<td>q₂</td>
<td>⊔</td>
<td>q_accept</td>
<td>⊔</td>
<td>L</td>
</tr>
<tr>
<td>q₂</td>
<td>a</td>
<td>q_reject</td>
<td>a</td>
<td>R</td>
</tr>
<tr>
<td>q₂</td>
<td>b</td>
<td>q₂</td>
<td>b</td>
<td>R</td>
</tr>
</tbody>
</table>

Give the sequence of configurations (as in class) resulting from executing the TM on each of the following strings.

(a). aabbb

(b). aabba

2. Design a TM to check for strings in the language \{w \mid w contains 3 0’s or 3 1’s in a row\} and give the corresponding transition table. If a given string is in the language, your TM should halt in the accepting state q_{accept}. Otherwise, your TM should halt in the rejecting state q_{reject}.

3. Design a TM to check for strings in the language \{w \mid w contains the same number of 1’s as 0’s\} and give the corresponding transition table. If a given string is in the language, your TM should halt in the accepting state q_{accept}. Otherwise, your TM should halt in the rejecting state q_{reject}. Hint: It might be helpful to “mark” symbols on the tape by writing something other than 0 or 1 (e.g., “x”).

4. Consider a TM for adding two binary numbers and provide a high-level description of the machine in plain English. Assume the input takes the form “b₁ # b₂” where b₁ is the first input string (binary
number), $b_2$ is the second input string (binary number), “#” separates the two numbers, and $b_1$ and $b_2$ each contain at least one digit. Your TM algorithm description should be broken into a series of numbered steps (e.g., “1. Scan to the # symbol.”) and you can use additional symbols as needed. As part of your algorithm, consider how the output will be captured on the tape.

5. State whether each of the following statements are true, false, or unknown, and provide a brief justification for each answer.

(a). If a problem is in P (PTIME) then it is also in NP (NPTIME).
(b). If a problem is in NP (NPTIME) then it is also in P (PTIME).
(c). If problem $A$ is in P and problem $B$ is NP-Complete, then $A \leq_P B$.
(d). If problem $A$ is in NP and problem $B$ is NP-Hard, then $B \leq_P A$.
(e). If problem $A$ is in NP and $B$ is in P, showing $A \leq_P B$ proves that P=NP.
(f). If problem $A$ is NP-Complete, $B$ is NP-Hard, and $B \leq_P A$, then $B$ is in NP.

6. Determine whether the following Boolean formula is satisfiable. If it is, give the corresponding variable assignment. If it isn’t, explain why not.

$$ (x \lor y) \land (\bar{x} \lor y) \land (x \lor \bar{y}) \land (\bar{x} \lor \bar{y}) $$

7. Consider the following Boolean formula. (a) Find a satisfying assignment for the formula, and (b) convert into a corresponding 3-SAT formula (using the approach from class).

$$ (x_1 \land (\bar{x}_2 \lor x_3)) \land (x_4 \lor \bar{x}_5) $$

8. Consider the following SUBSET-SUM problem in which we want to determine whether a subset sums to 5,842. Determine if a solution exists and if so give the corresponding certificate.

$$ \{493, 1153, 961, 1766, 1598, 1922, 1246, 1000, 869, 267\} $$

9. Convert the following 3SAT formula into a corresponding SUBSET-SUM problem using the approach described in class. Give the corresponding SUBSET-SUM problem (i.e., give $S$ and $t$) and solve it (by giving a certificate).

$$ \phi = (x_1 \lor \bar{x}_2 \lor \bar{x}_4) \land (x_2 \lor \bar{x}_3 \lor x_4) \land (\bar{x}_1 \lor x_3 \lor \bar{x}_4) $$

10. Consider the following one-person card game. A player picks $n^2$ cards from a deck and places them one-by-one on the table face up one row at a time such that there are $n$ rows and $n$ columns
of (face-up) cards. Each card is either blank, has a picture of a kitten, or has a picture of a puppy. (Note that if a row or column ends up with only blank cards, the corresponding row or column is removed.) The game proceeds by the player removing one kitten or puppy card at a time until each column contains only kitten or puppy cards (i.e., no column can have a mix of kitten and puppy cards), and each row contains at least one kitten or puppy card. In which case, the player wins. It may or may not be possible to win, depending on the configuration of cards drawn. Assume we want to develop an algorithm that takes a game configuration and determines whether or not it is winnable. Prove that this problem is NP-complete.

**Extra Credit.** It turns out that SAT solvers have a number of uses in real-world problems\(^1\). For a few extra credit points, follow along using the MiniSAT tutorial (below) to get the simple C++ program working in the tutorial (with Boolean variables \(A\), \(B\), and \(C\)) and provide a printout showing your program compile as well as the result of running your program. The tutorial is available here: [codingnest.com/modern-sat-solvers-fast-neat-underused-part-1-of-n/](http://codingnest.com/modern-sat-solvers-fast-neat-underused-part-1-of-n/)

---