EECS 376 Midterm Exam, Winter 2022

Multiple Choice – 30 points

Each question has one answer unless otherwise indicated. Fully shade in the circle with a pencil or black ink. Each question is worth 3 points.

1. Consider the following code. Note that the variables x and y are <u>real</u> numbers.

```
Require: x > y > 0 are real numbers                ▷ Hint: This actually matters...
function Foo376(x, y)
    if x ≤ 0 then return 1;
    z ← Foo376(x − log y, y)
    return (z + 1)
```

Which of the following is a valid potential function for the algorithm Foo376 (above)?

- ○ $s = x + y$
- ○ $s = e^y - x$
- ○ $s = x$
- ○ $s = x + 5y$
- ○ None of the above

2. In a new software company "Three-and-Out" a new "three-way" mergesort algorithm is used. The new algorithm splits the array into three equal parts instead of two equal parts, then sorts the three sub-arrays recursively and at the end merges the sorted sub-arrays. What is the correct recurrence relation for this new algorithm?

- ○ $T(n) = 2T(n/3) + O(n)$
- ○ $T(n) = 3T(n/2) + O(n \log n)$
- ○ $T(n) = 3T(n/3) + O(n)$
- ○ $T(n) = 2T(n/2) + O(n)$
- ○ $T(n) = 3T(n/2) + O(n)$
- ○ None of the above

3. Let $T(n) = 4T(n/3) + O(\sqrt{n})$. What is the tightest asymptotic upper bound for $T(n)$?

- ○ $O(n)$
- ○ $O(n^2)$
- ○ $O(\log n)$
- ○ $O(n \log n)$
- ○ $O(n^{\sqrt{2}})$
- ○ $O(2^n)$

4. Suppose $S_1 \subseteq \mathbb{R}$ is an uncountable set and $S_2 \subseteq \mathbb{R}$ is a countable set. Which of the following will always true?

   i Their intersection $S_1 \cap S_2$ is countable.

   ii Their union $S_1 \cup S_2$ is countable.

   iii The complement of S_1 is sometimes, but not always, countable.

   ◯ i and ii
   ◯ only i
   ◯ i and iii
   ◯ only iii
   ◯ ii and iii

5. A bottom-up dynamic programming algorithm fills the entire table, where the table has dimensions $n \times m$. What can be said about the runtime of the algorithm?

   ◯ $O(nm)$
   ◯ $\Theta(nm)$
   ◯ $o(nm)$
   ◯ $\Omega(nm)$
   ◯ No conclusions can be drawn from the information given.

6. Let L_1, L_2 be languages such that $L_1 \leq_T L_2$. If L_2 is undecidable, L_1 is ________ undecidable.

   ◯ always
   ◯ sometimes
   ◯ never

7. Consider the following algorithm for the 0-1 knapsack problem. Here n is the number of items, K is the total weight the knapsack can carry, and $V[i], W[i]$ are respectively the value and weight of the ith item. The output is the maximum total value of items that can be carried in the knapsack.

function KNAPSACK(n, K, V, W)
  if $n == 0$ or $K == 0$ **then return** 0
  if $W[n] > K$ **then return** KNAPSACK$(n-1, K, V, W)$
  else return $\max\{$ KNAPSACK$(n-1, K, V, W)$,
    KNAPSACK$(n-1, K - W[n], V, W) + V[n]$ $\}$

   What is the tightest bound on the worst-case running time of KNAPSACK?

   ◯ $O(n)$
   ◯ $O(n^2)$
   ◯ $O(nK)$
   ◯ $O(2^n)$
   ◯ $O(2^K)$

8. Which of the following languages (over ASCII) are decidable?

   i The set of all legal names of this term's EECS 376 IAs.
   ii The set of all prime numbers.
   iii The set of all valid natural deduction proofs.

   ◯ only i
   ◯ only i and ii
   ◯ only ii and iii
   ◯ none of i, ii, or iii
   ◯ all of i, ii, and iii

9. ~~Fill in the blank to make the following statement true and as strong as possible:~~

$$L \text{ is recognizable } \_\_\_\_\_\_\_\_ \ L \text{ is decidable.}$$

   ◯ If ($\Leftarrow$)
   ◯ Only if ($\Rightarrow$)
   ◯ If and only if ($\Leftrightarrow$)
   ◯ None of the above make the statement true

10. Which language is decidable? Assume M is a Turing Machine.

   ◯ $\{\langle M\rangle \mid M$ is a TM and $\exists M'$ such that $\epsilon \notin (L(M) \cap L(M'))\}$
   ◯ $\{\langle M\rangle \mid M$ is a TM such that $|L(M)|$ is odd$\}$
   ◯ $\{\langle M\rangle \mid M$ is a TM such that $0^{376k} \in L(M),$ for all $k \geq 1\}$
   ◯ $\{\langle M\rangle \mid M$ is a TM such that $x \notin L(M)$ for some $|x| = 376\}$
   ◯ $\{\langle M\rangle \mid M$ is a TM and $\exists M'$ such that $L(M) \cup L(M') = \{376\}\}$

UNIQNAME (print): ______________________________

Shorter Answer – 30 points

1. (8) Briefly prove or disprove the following statement:
   If $L_1, L_2 \subseteq \{0,1\}^*$ are undecidable languages and $L_1 \neq L_2$, then their symmetric difference, $(L_1 \setminus L_2) \cup (L_2 \setminus L_1)$, is also an undecidable language.

2. (6) Alice and Bob are competing to write a new algorithm to sort an array of size n. After some complexity analysis, they find that Alice's algorithm has time complexity $\Theta(n^2)$, while Bob's algorithm has time complexity $\Theta(n \log n)$. Bob expects his algorithm to run faster, but real-world testing on some moderately large arrays shows that Alice's algorithm actually runs faster. Briefly (ideally in two sentences or less each) answer the following questions:

   (a) Explain why this could be the case.

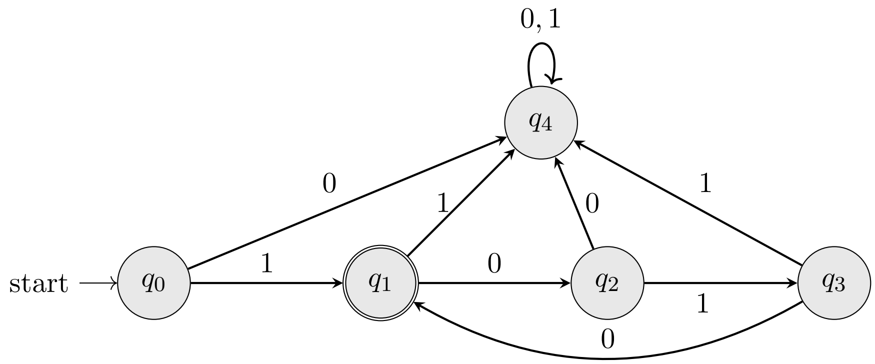
   (b) Can you be sure there exist inputs for which Bob's algorithm will run faster than Alice's? Justify your answer.

3. (6) A stay-put TM is an alternate model of TM. It is identical to the model of TM we learned in class (tape bounded on left, unbounded on right) except that the tape head is allowed to perform a "no-move." Namely, on each transition, the tape head is allowed to move one spot left, one spot right, or stay in place. Briefly answer the following questions (at a high level; one sentence per part is fine).

   (a) Can a stay-put TM simulate a normal TM?

   (b) Can a normal TM simulate a stay-put TM? If your answer is "yes," you only need to address how a normal TM would simulate writing a character and performing a "no-move." If you answer "no," you only need to provide a brief justification.

4. (10) All DFAs in this problem are over the alphabet $\Sigma = \{0, 1\}$.



(a) What is the language of the above DFA?

(b) Draw a DFA whose language is the set of all binary strings that begin with zero or more 0's followed by zero or more 01's (so $0^*(01)^*$). You should use 5 or fewer states.

Proofs and longer questions – 40 points

Answer two of the following three questions. **Clearly cross out the question you do not want graded.** If it isn't clear what problem you don't want graded, we will grade the first two. Each of the two questions are worth 20 points.

1. The **2-Arithmetic Subsequence Problem** is defined as follows: Given a sequence of n positive integers $a_1, \ldots, a_n$, find the length of the longest subsequence where each value in the subsequence is increasing by 2. For example, in the sequence $6, 1, 2, 3, 12, 4, 5, 6$, the length of the longest increasing-by-two subsequence is 3 and is achieved by the subsequence $2, 4, 6$ (also by $1, 3, 5$).

   (a) Find a recurrence relation that can be used to solve the 2ASP.

   (b) Write pseudocode which solves the problem using dynamic programming. It should have a run time that is $O(n^2)$.

2. Let $L_{bob} = \{\langle M\rangle \mid \langle\text{“bob”}\rangle \notin L(M)\}$. Show that $L_{ACC} \leq_T L_{bob}$ or show that $L_{HALT} \leq_T L_{bob}$. (Do whichever of the two you would prefer.)

3. Consider the following two-player game on a 12×12 grid. At the beginning of the game, there is at most one stone occupying each square (so each square either has 1 or 0 stones). On their turn, each player must pick a stone and remove it. Subsequent to removing the stone, the player can place or remove stones as desired, on any of the squares to the **right** of that stone (on the same row). No square may ever have more than one stone. The game ends when there are no more stones left on the board.

   Are there any initial board states **and** sets of moves that can make the game continue indefinitely, or will the game always end **regardless** of the initial position and the moves of the players? If so, describe them. If not, provide a proof that demonstrates this.

   Hint: Try simulating the game on a 4×4 grid.

This page intentionally left blank.

If you have answers on this page be certain you write "answer continues on page 12" under the actual question.