CSC4010 (Practice Midterm) **Date: 22 October 2026**
Name:

Instructions: This exam is closed book, closed notebook. No electronic devices are allowed. You may use one A4-size study sheet (one-sided) that you prepared yourself. The last few pages of the exam are scratch paper; you may not use your own. **Please print your ID number at the top of each page.**

Any deviation from these rules may lead to cancellation of your exam.

The exam consists of 3 multiple-choice, 4 short-answer, and 2 longer-answer questions, for a total of 36 points. For the short- and longer-answer sections, please write your answers clearly in the spaces provided. If you run out of room or need to start over, you may use the blank pages at the end, but you MUST make that clear in the space provided for the answer. The exam has 4 pages printed on both sides, including this page and the blank pages at the end.

Multiple Choice: Select <u>all</u> valid options (4 points each)

For each of the problems in this section, select <u>all</u> valid options; this could be all of them, none of them, or something in between.

A fully correct solution earns all 4 points. Each error (selection or non-selection) reduces your score by 1 points (to a minimum of 0 points).

1. (4 points) Select all of the **countable** sets.

   ☐ $\{\langle M\rangle : M \text{ is a Turing Machine}\}$
   ☐ $\{\langle M\rangle x : M \text{ is a Turing Machine and accepts } x\}$
   ☐ $\{L : L \subseteq \{0,1\}^*\}$
   ☐ $\{L : L \subseteq \{0,1\}^* \text{ is a finite language}\}$

2. (4 points) Select each hypothesis that implies that L is decidable.

   ☐ Language $\bar{L}$ is decidable.
   ☐ Language A is decidable and $L \subseteq A$.
   ☐ Language $A \leq_m L$ and A is decidable.
   ☐ Language $L = A \cup B$, and at least one of A or B is decidable.

3. (4 points) Consider three algorithms with the following worst-case running times:

$$T_A(n) = O(n^3)$$
$$T_B(n) = \Theta(n^2)$$
$$T_C(n) = \Theta(n \log n)$$

   Recall that the worst-case running time represents the maximum number of steps an algorithm takes over all inputs of size n. **Select all statements that are necessarily true based on the given information.**

   ☐ Algorithm A has worst-case running time $\Omega(n^2)$
   ☐ Algorithm B has worst-case running time $O(n^3)$
   ☐ Algorithm C runs faster than Algorithm B for all inputs.
   ☐ For all large enough n, there exists an input of size n for which C runs faster than B.

Short Answer (3 points each)

1. (3 points) Consider the set of all bijections $f : \mathbb{N} \to \mathbb{N}$. Is this set countable or uncountable? Give a short explanation.

2. (3 points) Consider a variant of a Turing machine with a two-way infinite tape. At each step, its head reads and writes only the current cell and then moves exactly two cells to the left or right, without reading or writing the skipped cell. The binary input is written in consecutive cells, and the head initially points to the first input cell. Can this variant decide every decidable language over $\{0, 1\}$? Give a short explanation.

3. (3 points) Let $A, B \subseteq \{0,1\}^*$ be recognizable languages. Show that $A \cup B$ is recognizable.

4. (3 points) Let $\mathrm{COMPOSITE}$ be the language of nonempty binary strings representing composite integers. Briefly describe a polynomial-size certificate and a polynomial-time verifier showing that $\mathrm{COMPOSITE} \in \mathrm{NP}$.

Long Answer Question (6 points each)

1. (6 points) Let $L_{\text{even-acc}} = \{\langle M \rangle \mid \forall x \in \{0,1\}^*,\ M(x) \text{ accepts } \iff |x| \text{ is even}\}$. That is, let $L_{\text{even-acc}}$ be the language of descriptions of all Turing machines that accept exactly the set of even-length strings. Show that this language is unrecognizable.

2. (6 points) Consider the language $L = \{\langle M \rangle \mid L(M)$ contains a string of the form $0^n 1^n$ for $n \geq 0\}$. Is this language decidable or undecidable? Is it recognizable or unrecognizable? Justify your answer.

Solutions

Multiple Choice

1. Consider each of the options:

   - $\{\langle M\rangle : M \text{ is a Turing Machine}\} \subseteq \{0,1\}^*$, which is countable.
   - $\{\langle M\rangle x : M \text{ is a Turing Machine and accepts } x\} \subseteq \{0,1\}^*$ which is countable.
   - $\{L : L \subseteq \{0,1\}^*\} = \mathcal{P}(\{0,1\}^*)$, this implies by Cantor's theorem that $|\{L : L \subseteq \{0,1\}^*\}| > |\{0,1\}^*| = |\mathbb{N}|$.
   - $\{L : L \subseteq \{0,1\}^* \text{ is a finite language}\}$ is a countable set. We can find an ordering of its elements as follows: first, classify them into categories according to the total length of the strings they contain. Each category is a finite set, whose size we can determine. So, we can order sets within each category according to an ordering of our choice. Finally, we order each of the categories according to the increasing total length of the strings they contain.

2. Consider each of the following options:

   - If $\bar{L}$ is decidable, then L is also decidable. This is because we can simply take the decider for $\bar{L}$ and reverse its output to obtain a decider for L.
   - This statement does not imply L is decidable. Consider the language $A = \{0,1\}^*$. This is decidable, because a Turing machine that accepts all strings is a decider for this language, and any language L, including the undecidable ones we did in class are a subset of A.
   - Reducibility to a decidable language gives decidability; reducibility from one does not. So, this statement does not imply decidability of L.
   - This does not imply decidability of L. Consider the trivial case where $A = A_{TM}$ and $B = \emptyset$.

3. Consider each of the following options:

   - The fact that $T_A(n) = O(n^3)$ does not rule out the possibility that $T_A(n) = \Theta(n \log n)$. So, this statement is not necessarily true.
   - This statement is true: $T_B(n) = \Theta(n^2)$, implies there exists $c, n_c > 0$ such that for all $n > n_c$, $T_B(n) \leq cn^2 \leq cn^3$. Thus, $T_B(n) = O(n^3)$.
   - This statement is not necessarily true: Worst-case bounds do not determine which algorithm is faster on every individual input. For example, B could take linear time on some inputs while C could take time $\Theta(n \log n)$ on those inputs.
   - This statement is necessarily true: For sufficiently large n, some input makes B take cn^2 steps, while C takes time at most $dn \log n$ on every input of that size. Since $dn \log n < cn^2$ eventually, C is faster on that input.

Short Answer

1. **The set is uncountable. The tricky part here is that we also need the diagonal function to be a bijection.** Take $\mathbb{N} = \{1, 2, \ldots\}$. Suppose, for contradiction, that all bijections $\mathbb{N} \to \mathbb{N}$ can be listed as $f_1, f_2, \ldots$.

   Define $g : \mathbb{N} \to \mathbb{N}$ by

$$\big(g(2i-1), g(2i)\big) = \begin{cases} (2i,\ 2i-1), & \text{if } f_i(2i-1) = 2i-1, \\ (2i-1,\ 2i), & \text{otherwise.} \end{cases}$$

Thus, g either swaps or fixes the elements of each pair $\{2i-1, 2i\}$, so g is a bijection.

By construction, for every i,

$$g(2i-1) \neq f_i(2i-1).$$

Therefore, g differs from every bijection in the list, contradicting the assumption that the list contains all bijections. Hence the set is uncountable.

2. No. Every move preserves the parity of the head's position. Starting at the blank cell right before the first input cell, the head can access only the second, fourth, sixth,$\cdots$ input symbols; it can never read the first symbol For example, the inputs 10 and 00 have identical contents in every accessible cell. Therefore, the machine must behave identically on both inputs. But the finite—and hence decidable—language $L = \{10\}$ requires accepting 10 and rejecting 00. Thus this variant cannot decide every decidable language.

3. Let M_A, M_B be recognizers for A, B respectively, the recognizer M for $A \cup B$ does the following: on input x, it alternates between running $M_A(x)$ and $M_B(x)$ for one step. If at some point either M_A or M_B accept x, then M accepts. If $x \in A \cup B$, then either M_A or M_B accept x. WLOG this happens in t steps. Then, in $2t$ steps, M will accept x as well. Thus, M is a recognizer for $A \cup B$.

4. Let N be the number represented by the input string $x \in \{0,1\}^*$. The **certificate** is a number d such that $1 < d < N$, and d divides N. Note that the size of the certificate is at most $|x|$. The **verifier** checks if: a) If $1 < d < N$. b) d divides N without leaving any remainders. Binary comparison and division take polynomial time. Moreover, if N is not composite then no such certificate exists.

Long Answer

1. Consider the following reduction mapping reduction f from $\bar{A}_{TM}$ to $L_{\text{even-acc}}$. Note that $\bar{A}_{TM} = \{\langle M, w\rangle \mid M \text{ does not accept } w\}$, and $f(\langle M, w\rangle) = \langle M_w\rangle$. TM $\langle M_w\rangle$ on input x does the following:

   a) If $x \neq 1$ and has odd length, then reject,

   b) else if, x has even length, then accept,

   c) else if $x = 1$ then simulate M on w.

   Note that $\langle M_w\rangle \in L_{\text{even-acc}} \iff \langle M, w\rangle \in \bar{A}_{TM}$

2. It is undecidable and recognizable. The recognizer will do dovetailing: it will run in phases in ith phase, it will continue the run of M on strings $\varepsilon, 01, \cdots, 0^i1^i$ for one step each. If in a phase, any of the strings are accepted by M, then the recognizer accepts. Otherwise, it continues its run. If there exists $0^n1^n \in L(M)$ that accepts in t steps, then M will be first simulated on it in phase n, and at the end of phase $n+t$, it will be accepted. To see undecidability, consider the following mapping reduction from A_{TM} to L: $f(\langle M, w\rangle) = \langle M_w\rangle$, where M_w on input x does the following:

   1. If $x \neq 01$, then reject.
   2. Else, run M on w. If M accepts, then accept 01.

   $\langle M_w\rangle \in L \iff \langle M, w\rangle \in A_{TM}$.