Theory of Computation

Spring 2025, Final Exam.

2:20-5:00 PM, June 3, 2025

In this test, we write $\leq_m$ for many-one reduction, $\leq_P$ for polynomial-time reduction, and $\leq_L$ for logspace reduction. We also write *r.e.* for recursively enumerable languages, P for the class of languages in deterministic polynomial time, and NP for the class of languages in nondeterministic polynomial time. Given two languages L_1 and L_2, $L_1 \setminus L_2 = \{w \mid w \in L_1, w \notin L_2\}$.

1. (40 pts) True or False? No penalty for wrong answers. No explanations are needed.

   (1) The language $L = \{a^i b^j c^k \mid j < i, j < k\}$ is not context-free.
   Sol. True. Easy proof using pumping lemma.

   (2) The complement of the language $L = \{a^i b^i c^i d^i \mid i \geq 0\}$ is context-free.
   Sol. True. L consists of (1) strings not of the form $a^*b^*c^*d^*$ which can be checked using an FA, and (2) the lengths of two of a^i, d^i, c^i d^i are different, which can be checked by a PDA.

   (3) If both L and $\overline{L}$ are context-free, then L must be regular.
   Sol. False. $\{a^n b^n \mid n \geq 0\}$ and its complement are both context-free.

   (4) If both L_1 and L_2 are deterministic context-free languages, then $L_1 \cap L_2$ is a context-free (not necessarily deterministic) langauge.
   Sol. False. $\{a^n b^n c^m \mid m, n \geq 0\} \cap \{a^n b^m c^m \mid m, n \geq 0\}$.

   (5) If both L_1 and L_2 are context sensitive languages, then $L_1 \cap L_2$ is also context-sensitive.
   Sol. True. Context-sensitive languages are those that can be accepted by linear bounded automata.

   (6) If L_1 and $L_1 \cap L_2$ are context-free, then L_2 must be regular.
   Sol. False. $L_1 = \{a^n b^n\}$ and $L_2 = \{0^n 1^n\}$.

   (7) Given two context-free languages L_1 and L_2, the problem "$L_1 \cap L_2 = \emptyset$?" is decidable.
   Sol. False. $\{\langle M, w\rangle \mid M \text{ is a TM and } w \in L(M)\}$ can be expressed as the intersection of two CFLs whose intersection represents sequences of accepting computations.

   (8) Given a linear bounded automaton M and a regular language R, deciding whether $L(M) \subseteq R$ is decidable.
   Sol. False. $L(M) \subseteq \emptyset$ iff $L(M) = \emptyset$, which is undecidable.

   (9) Given two r.e. languages L_1 and L_2, $L_1 \setminus L_2$ (containing words in L_1 but not in L_2) is always r.e.
   Sol. False. $\overline{A_{TM}} = \Sigma^* \setminus A_{TM}$.

   (10) All regular languages are in the class DSPACE(1), i.e., accepting by DTMs using constant work space.
   Sol. True. FAs are TMs not using extra space.

   (11) The class P is closed under complementation.
   Sol. True. Let M be a DTM accepting a language in P. Swapping the accepting and rejecting states of M accepts the complement of the language.

   (12) The class NP is closed under the star " $*$ " operation. (That is, $L \in NP \Rightarrow L^* \in NP$).
   Sol. True. On input $w \in \Sigma^*$, guess a partition $w = w_1 w_2 ... w_k$, $\forall i, w_i \in \Sigma^*$. Check $w_i \in L$.

   (13) The language $L = \{\langle M_1, M_2\rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) \subseteq L(M_2)\}$ is not r.e.
   Sol. True. Let $L(M_2) = \emptyset$. Then $E_{TM} \leq_m L$, where $E_{TM} = \{\langle M\rangle \mid L(M) = \emptyset\}$ is known to be not in r.e.

   (14) Given a context-free grammar G and a string w, deciding whether $w \in L(G)$? is solvable in P.
   Sol. True. CYK algorithm runs in P-time.

   (15) The undecidability of $\{\ \langle M\rangle \mid M$ is a minimal TM, i.e., no TM with a smaller representation recognizes the same language $\}$ follows from Rice's theorem.
   Sol. False. It is not a TM property.

(16) $A_{TM} \leq_m 0^*1^*$.
Sol. False. 0^*1^* is regular, which is recursive.

(17) If $A \leq_P B$ and A is recursive, then B must be recursive.
Sol. False. $\{ \langle M, w, n \rangle \mid M$ halts on w in n steps $\} \leq_P \{ \langle M, w \rangle \mid M$ halts on w $\}$. The former is recursive while the latter is r.e. but not recursive.

(18) $SAT \leq_P A_{TM}$.
Sol. True. Given a formula ϕ, maps to a $\langle M, \phi \rangle$ such that M guesses a truth assignment and checks whether it satisfies ϕ. If yes, accept.

(19) The language $\{\langle M \rangle \mid M$ is a TM that does not accept string 0101$\}$ is in r.e. but not recursive.
Sol. False. The language is not in r.e., using strong Rice's theorem.

(20) Given a deterministic context-free language (DCFL) L and a regular language R, $L \cap R$ is a DCFL.
Sol. True. L can be accepted by a DPDA, and R by a DFA. The product of the two automata is a DPDA accepting $L \cap R$.

2. (25 pts) Give a brief yet convincing argument for each of the following statements.

   (a) Let $A_{NFA} = \{\langle M, w \rangle \mid M$ is an NFA and $w \in L(M)\}$. A_{DFA} is defined similarly. It is known that A_{NFA} is in $NSPACE(\log n)$. Show that $A_{NFA} \leq_P A_{DFA}$.
   Sol. Though we cannot convert an arbitrary NFA to an equivalent DFA in poly time, since the DFA might have exponentially many states, we don't need to do that. Given input $\langle M, w \rangle$, we can determine in P (since $NSPACE(\log n) \subseteq P$) whether $\langle M, w \rangle$ is in A_{NFA} (which is basically the PATH problem discussed in class), then output either a fixed string in A_{DFA} or a fixed string not in A_{DFA}.

   (b) Knowing that A_{DFA} is in $DSPACE(\log n)$ (deterministic logspace), show that there exists a context free language X such that $A_{DFA} \leq_L X$.
   Sol. Let $X = \{1\}$, let the reduction compute whether $\langle M, w \rangle$ is in A_{DFA} (using deterministic logspace to simulate M on w) and then output 1 or 0 according to the answer.

   (c) Given two context-free languages L_1 and L_2, show that $L_1 \setminus L_2$ is always recursive.
   Sol. L_1 and L_2 are both recursive. Let M_1 and M_2 be TMs accepting L_1 and L_2, resp. On input w, M simulates M_2 on w. If M_2 accepts, then M rejects. If M_2 rejects, then M starts the simulation of M_1 on w. If M_1 accepts, then M also accepts; otherwise M rejects. It is easy to show that M always halts.

   (d) The language $L = \{\langle M, w, q \rangle \mid M$ is a DTM and M never visits state q on input $w\}$ is not recursive.
   Sol. $A_{TM} \leq \overline{L}$ such that $f(\langle M, w \rangle) = (\langle M', w, q \rangle)$, where M' is M by adding an ϵ transition of M to a new state q. M accepts w iff M' eventually enters state q.

   (e) Suppose that there is an L_1 which is NP-complete, and $\overline{L_1} \in$ NP. Prove that for all $L \in$ NP, it must be the case that $\overline{L} \in$ NP.
   Sol: For all $L \in$ NP, $L \leq_P L_1$, hence, $\overline{L} \leq_P \overline{L_1}$. Since $\overline{L_1} \in$ NP, $\overline{L}$ is in NP.

3. (15 pts) There are five languages A, B, C, D, E. All we know about them is the following: (1) A is in P. (2) B is in NP. (3) C is NP-complete. (4) D is Recursive. (5) E is r.e. but not Recursive. For each of the five statements (a) - (e) below,
(a) $E \leq_m D$ (b) $C \leq_P B$ (c) $A \leq_P B$ (d) $B \leq_P \overline{C}$ (the complement of C) (e) $D \leq_m C$
tell whether it is:

   - **CERTAIN** to be true, regardless of what languages A through E are and regardless of the resolution of unknown relationships among complexity classes, of which "is P = NP?" is one example.
   - **MAYBE** true, depending on what languages A through E are, and/or depending on the resolution of unknown relationships such as P = NP?
   - **NEVER** true, regardless of what A through E are and regardless of the resolution of unknown relationships such as P = NP?

No explanation needed.
Sol: (a) NEVER. (b) MAYBE. (c) MAYBE. (d) MAYBE. (e) CERTAIN.
(Reason)

(a) If $E \leq_m D$, knowing D in recursive implies E is recursive – a contradiction.

(b) If B is NP-C, $C \leq_P B$ holds. If B is in P and P $\neq$ NP, $C \leq_P B$ implies C in P - contradicting P $\neq$ NP.

(c) If B is NP-C, $A \leq_P B$ holds. If B is the language $\emptyset$ (which is clearly in NP), A is a^*b^*, $A \leq_P B$ does not hold as there is no way to map a^*b^* to the empty set.

(d) If B is in P, $B \leq_P \overline{C}$ holds. If B is NP-C and NP $\neq$ co-NP, then $B \leq_P \overline{C}$ does not hold.

(e) Let M be a TM that decides D. Let C = 3SAT. Using M to define $f(w) = (x \vee y \vee z)$ (a satisfiable formula) if $w \in D$; $f(w) = (x \wedge x \wedge \overline{x})$ (a non-satisfiable formula) if $w \notin D$. Hence, $D \leq_m C$ holds.

4. (20 pts) Let $A_R = \{\langle M \rangle \mid M \text{ is a TM and } L(M) = L(M)^R\}$. Note that $L(M)^R = \{w^R \mid w \in L(M)\}$, where w^R is the reversal of w. Note that TMs accepting $\emptyset$ and Σ^* are both in R_{TM}. Fill in each of the blanks below.

(a) To prove $A_{TM} \leq_m A_R$, we use the following mapping from $\langle M, w \rangle$ to $\langle M' \rangle$:

Given M and w, construct M' = "On input x"
(1) If $x = 01$, $\boxed{\text{① ....ACCEPT............}}$
(2) Run M on w
(3) If $\boxed{\text{② .....}M \text{ accepts } w \text{.....................}}$, then $\boxed{\text{③ ....ACCEPT............}}$

- $A_{TM} \leq_m A_R$ holds because if M accepts w, $L(M') = \boxed{④ ....\Sigma^*.............}$; otherwise, $L(M') = \boxed{⑤ ....\{01\}.............}$.

(b) To prove $\overline{A_{TM}} \leq_m A_R$, we use the following mapping from $\langle M, w \rangle$ to $\langle M' \rangle$:

Given M and w, construct M' = "On input x"
(1) Run M on w
(2) If $x = 01$ and $\boxed{\text{⑥ ....}M \text{ accepts } w \text{ ....}}$, then $\boxed{\text{⑦ ....ACCEPT...........}}$

- $\overline{A_{TM}} \leq_m A_R$ holds because if M accepts w, $L(M') = \boxed{⑧ ....\{01\}............}$; otherwise, $L(M') = \boxed{⑨ ....\emptyset.............}$.

Based on $A_{TM} \leq_m A_R$ and $\overline{A_{TM}} \leq_m A_R$, write the largest language class that A_R is NOT in $\boxed{\text{⑩ .... (r.e. ∪ co-r.e.) ....}}$. (Choose from one of the following: NP, co-NP, r.e., co-r.e., recursive, (r.e. ∪ co-r.e.), (r.e. ∩ co-r.e).)