Theory of Computation

Spring 2023, Final Exam (June 6, 2023)

1. (15 pts) True or False? Give a short yet convincing argument.

   (1) Let f be any morphism from Σ^* to Σ^*. Then for any language R over Σ, R is a context-free language if and only if $\{f(w) : w \in R\}$ is a context-free language.
   Sol. False. Consider $f(a) = f(b) = \epsilon$. Then $f(\{a^n b^n a^n \mid n \geq 1\} = \{\epsilon\}$, which is CF. But $\{a^n b^n a^n \mid n \geq 1\}$ is not CF.

   (2) Given a context-free language L and a regular language R, checking "$L - R = \emptyset$?" is decidable. The "$-$" operator denotes the set difference.
   Sol. True. $L - R = L \cap \overline{R}$, which is CF. It is decidable whether a CFL is empty or not. @

   (3) $A_{TM} \leq_m \{0^n 1^n | n \geq 0\}$, where $\leq_m$ stands for the many-one reduction.
   Sol. False. $\{0^n 1^n | n \geq 0\}$ is CF, which is a recursive language. If the reduction holds, then A_{TM} is recursive, which is a contradiction.

   (4) If *co-NP* $\subseteq NP$, then *co-NP* $= NP$. Here *co-NP* is the complement of NP.
   Sol. True. Consider an $X \in NP$, then $\overline{X} \in co-NP$ (by definition), and hence, $\overline{X} \in NP$ (since *co-NP* $\subseteq NP$). Therefore, $X \in co-NP$ (by definition). We thus have $X \in NP \Rightarrow X \in co-NP$, which implies $NP \subseteq co-NP$.

   (5) The language $\{ \langle M, x, 1^t \rangle \mid M$ is a deterministic TM which accepts x in t steps $\}$ is in P (i.e., polynomial time).
   Sol. True. Design a DTM M', when given $M, x, 1^t$ as its input, simulate M on x (in a way like a universal TM) for at most t steps and accept if $M(x)$ accepts within that time bound.

2. (15 pts) Consider the following grammar G in Chomsky Normal Form:

$$S \to a \mid YZ; \qquad Z \to a \mid ZY; \qquad Y \to b \mid ZZ \mid YY$$

   where S is the start symbol. Use the CYK (dynamic programming) algorithm to determine whether the string *babba* is in $L(G)$ or not. To this end, we define X_{ij} to be the set of nonterminals that can derive the substring from positions i to j, where $i \leq j$. For example, the substring from positions 2 to 4 is *abb*, and from positions 4 to 5 is *ba*. Complete those X_{ij} entries in the following table:

X_{ij}	1	2	3	4	5
1	Y	S	S	S	Y
2	-	S, Z	Z	Z	Y
3	-	-	Y	Y	S
4	-	-	-	Y	S
5	-	-	-	-	S, Z

   Hence, $babba \notin L(G)$.

3. (20 pts) Consider the following language classes numbered as (C1)-(C7).
   (C1) Empty; (C2) Finite (FI); (C3) Regular (REG); (C4) Context-free (CF); (C5) Recursive (rec); (C6) Recursively enumerable (r.e.); (C7) All languages.
   For each of the following languages, give the *lowest numbered* class to which it surely belongs. No explanations needed. For example, if a language is context-free but not regular, then answer "C4".

   (1) $L_1 - L_2,\ L_1, L_2 \in CF$ (2) $L_1 - L_2,\ L_1, L_2 \in rec$ (3) $L_1 - L_2,\ L_1, L_2 \in r.e.$
   (4) $L_1 - L_2,\ L_1 \in r.e., L_2 \in FI$ (5) $L_1 - L_2,\ L_1 \in REG, L_2 \in r.e.$ (6) $\{(a^i b^i)^i \mid i \geq 1\}$
   (7) $\overline{L},\ L \notin rec$ (8) $L_1 \cap L_2,\ L_1 \in rec, L_2 \in CF$ (9) $\overline{L},\ L \in CF$
   (10) $L^*,\ L \in rec$

   Sol. (1) C5 (Consider $\Sigma^* - \overline{\{a^n b^n c^n \mid n \geq q\}}$ (2) C5 ($L_1 - L_2$ is also recursive) (3) C7 (Consider $\Sigma^* - A_{TM}$) (4) C6 ($L_1 - L_2$ is also r.e.) (5) C7 (Consider $\Sigma^* - A_{TM}$) (6) C5 (the language is recursive but not CF) (7) C7 (Consider EQ_{TM}) (8) C5 ($L_1 \cap L_2$ is also recursive) (9) C5 (Consider $\{a^n b^n c^n \mid n \geq q\}$ whose complement is CF) (10) C5 (rec is closed under Kleene star).

4. (25 pts) For each of the following languages, decide whether it is decidable or not. Give a convincing argument. To show a language to be undecidable, do not use Rice's theorem. TMs are assumed to be deterministic.

   (a) $L_1 = \{\langle M_1, M_2 \rangle \mid |L(M_1) \cap L(M_2)| = 2\}$, i.e., the set of pairs of TMs such that the intersection of their languages contains exactly two strings.
   Sol. Undecidable. Given $\langle M, x \rangle$,

   - M_1: on input y, run M on x, accepts if M accepts x;
   - M_2: a TM that accepts two strings.

   Clearly, $\langle M, x \rangle \in A_{TM}$@iff $|L(M_1) \cap L(M_2)| = 2$, i.e., $\langle M_1, M_2 \rangle \in L_1$.

(b) $L_2 = \{\langle M \rangle \mid M$ halts on at least one input $\}$.
Sol. Undecidable. Given $\langle M, x\rangle$,

- M': on input y, run M on x.

Clearly, $\langle M, x\rangle \in HALT_{TM}$ iff $\langle M'\rangle \in L_2$.

(c) $L_3 = \{\langle M, x\rangle \mid M$ reads from more than 100 tape cells in accepting input $x\}$.
Sol. Undecidable. Given $\langle M, x\rangle$,

- M', ϵ: move right to read 100 symbols and then reset its head; then run M on x, accept if M accepts x.

Clearly, $\langle M, x\rangle \in A_{TM}$ iff $\langle M', \epsilon\rangle \in L_3$.

(d) $L_4 = \{\langle M \rangle \mid M$ never enters a specific state on any input $\}$.
Sol. Undecidable. Given $\langle M, x\rangle$,

- M': on input y, run M on x, if M accepts x then enters a specific new state s not in M.

Clearly, $\langle M, x\rangle \in A_{TM}$ iff $\langle M'\rangle \in \overline{L_4}$.

(e) $L_5 = \{\langle M \rangle \mid M$ enters some state more than 100 times when give a blank tape $\}$.
Sol. Decidable. Suppose M has n states. Simulate M on the blank tape,

- Case 1: M halts before $100n$ steps.
  if M enters some state more than 100 times, then answer "YES"; otherwise answer "NO".
- Case 2: M does not halt after $100n$ steps, answer "YES".

5. (15 pts)

(a) (7 pts) Consider *UNARY-PCP* which is the subset of PCP where the instance's alphabet Σ has a single letter. That is, given $\{(x_1, y_1), ..., (x_k, y_k)\}$, where $x_i, y_i \in \{a\}^*$, decide whether there exists $i_1, i_2, ..., i_h$ such that $x_{i_1}x_{i_2}\cdots x_{i_h} = y_{i_1}y_{i_2}\cdots y_{i_h}$. Is Unary-PCP decidable? Justify your answer.
Sol. Given a set of dominoes with a string in a^* on the top and another string in a^* on the bottom, i.e., of the form (a^i, a^j), we decide whether this set is in PCP as follows.

- If a domino of type (a^i, a^i) exists, answer "YES";
- If every domino of type (a^i, a^j) has $i > j$ (or $i < j$), answer "NO";
- Otherwise, there must exist two domino types (a^{b+c}, a^b) and (a^d, a^{d+e}) with $c, e > 0$. As e copies of the first domino and c copies of the second give up a match because there are $be + ce + cd$ a's in the top string and the bottom string. Thus, answer "YES".

(b) (8 pts) Consider *Binary-PCP* which is the subset of PCP when $\Sigma = \{0, 1\}$. Is Binary-PCP decidable? Justify your answer.
Sol. Suppose the original PCP P is over alphabet $\{a_1, a_2, ..., a_n\}$. We define a PCP P' over alphabet $\{0, 1\}$ in the following way. PCP P' is constructed from P by replacing symbol a_i by 10^i. For example, a pair $(a_3a_2, a_1a_4a_2)$ maps to $(1000100, 1010000100)$.

6. (10 pts) Suppose we want to use the pumping lemma to prove that $L = \{x \in \{a, b\}^* \mid N_a(x) < N_b(x) < 2N_a(x)\}$ is not context-free, where $N_a(x)$ (resp., $N_b(x)$) is the number of a's (resp., b's) in x. The proof starts with picking the word $s = a^{p+1}b^{2p+1} \in L$, where p is the pumping constant. Now s can be partitioned into $uvxyz$ meeting the conditions of the pumping lemma. The arguments for the cases $vxy = a^j$, $vxy = b^j$, and v (or y) contains a mixture of a and b are quite simple. To complete the proof, show <u>in detail</u> when vxy contains both a's and b's. (Hint: consider the two cases $N_a(vy) > N_b(vy)$ and $N_a(vy) \le N_b(vy)$.)
Sol.

- If $N_a(vy) > N_b(vy)$ then we can pump down to get uxz, for which $N_a(uxz) = p + 1 - j$ and $N_b(uxz) = 2p + 1 - k$ with $j < k$. But $2(p + 1 - j) = 2p + 1 + 1 - 2j < 2p + 1 - k$, since $1 - 2j < -k$ whenever $j < k$, so uxz is not in L.
- If $N_a(vy) \le N_b(vy)$, on the other hand, we can pump up to get a number of b's more than the number of a's: if we use the string uv^pxy^pz, $N_a(uv^pxy^pz) = p + 1 + (p - 1)j$ and $N_b(uv^pxy^pz) = 2p + 1 + (p - 1)k$ where $k \ge j$. Then $2N_a(uv^pxy^pz) = 2p + 2 + 2j(p - 1) < 2p + 1 + (p - 1)k$.