CS 341-451, Fall 2021, eLearning (online) Section
Solutions for Midterm 1

1. (a) True. If A has an NFA, then it is regular, and all regular languages are context-free.

   (b) False. Suppose that A is a nonregular language defined over an alphabet Σ. Let $B = \overline{A}$ be the complement of A, so $B = \Sigma^* - A$. We must have that B is also nonregular because if B were regular, then $\overline{B}$ would also be regular, but $\overline{B} = A$, which we assumed is nonregular. Now note that $A \cup B = A \cup \overline{A} = \Sigma^*$, which is regular.

   (c) False. Let $A = \{ a^n b^n c^n \mid n \geq \}$ and $B = \{ c^n b^n a^n \mid n \geq \}$, which are both non-context-free. Note that $A \cap B = \{\varepsilon\}$, which is finite, so the intersection is regular, which implies that it is also context-free.

   (d) False. The language a^* is regular but infinite.

   (e) True. Suppose that language A is Turing-decidable, and we want to prove that its complement $\overline{A}$ is also Turing-decidable. Because A is Turing-decidable, there is a TM M that decides A. Specifically, M accepts each string $w \in A$, and M rejects each string $w \notin A$, so M never loops. Now define another TM M' to be the same as M but with the accept and reject states swapped. Now M' accepts each string $w \notin A$, and M' rejects each string $w \in A$, and M' never loops. Thus, M' decides $\overline{A}$, so $\overline{A}$ is decidable.

   (f) False. HW 6, problem 2(a).

   (g) False. The language A is non-context-free, which can be proven using the same basic proof on slides 2-96 and 2-97, so A cannot have a CFG.

   (h) True. To verify this, we need to show that every Turing-decidable language is also Turing-recognizable. Suppose that A is Turing-decidable. Then there is a TM M that decides A, so M also recognizes A. Thus, A is also Turing-recognizable.

   (i) False. The language $A = \{ a^n b^n c^n \mid n \geq 0 \}$ is nonregular. But A is also non-context-free (slides 2-96 and 2-97), so A cannot have a context-free grammar.

   (j) True. By Kleene's theorem, the class of languages having regular expressions is the class of regular languages, which is closed under concatenation by Theorem 1.26.

2. (a) $b^*(ba^*b \cup a)ab^*$. Other regular expressions for the language include $b^*ba^*bab^* \cup b^*aab^*$ and $b^*(ba^*b \cup a)ab^* \cup \emptyset$. There are infinitely many correct regular expressions for the language.

   (b) $G_3 = (V_3, \Sigma, R_3, S_3)$ with $S_3 \notin V_1 \cup V_2$, where

   - $V_3 = V_1 \cup V_2 \cup \{S_3\}$,
   - S_3 is the (new) starting variable,
   - Σ is the same alphabet of terminals as in G_1 and G_2, and
   - $R_3 = R_1 \cup R_2 \cup \{S_3 \to S_2 S_1\}$.

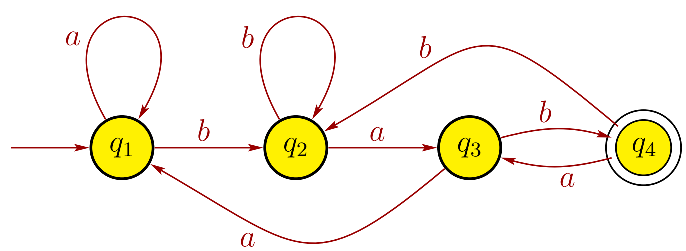
(c) After the one step of removing $A \to \varepsilon$, the CFG is then

$$\begin{aligned} S_0 &\to S \mid \varepsilon \\ S &\to ASA0A \mid AA0A \\ A &\to 0SA \mid 0A \mid 0SA1S01A \mid 0A1S01A \mid 0SA101A \mid 0A101A \mid \varepsilon \end{aligned}$$

(d) $M_3 = (Q_3, \Sigma, \delta_3, q_3, F_3)$, where

- $Q_3 = Q_1 \times Q_2$;
- Σ is the same alphabet as M_1 and M_2 have;
- the transition function δ_3 satisfies $\delta_3((q,r),\ell) = (\delta_1(q,\ell), \delta_2(r,\ell))$ for $(q,r) \in Q_3$ and $\ell \in \Sigma$;
- the starting state $q_3 = (q_1, q_2)$; and
- $F_3 = (Q_1 \times F_2) \cup (F_1 \times Q_2)$

3. (a) A DFA for $C = \{\, w \in \Sigma^* \mid w = sbab \text{ for some } s \in \Sigma^* \,\}$, $\Sigma = \{a, b\}$, is below:



A 5-tuple description of the DFA above is $M = (Q, \Sigma, \delta, q_1, F)$, where

- $Q = \{q_1, q_2, q_3, q_4\}$
- $\Sigma = \{a, b\}$
- The transition function $\delta : Q \times \Sigma \to Q$ is defined as

	a	b
q_1	q_1	q_2
q_2	q_3	q_2
q_3	q_1	q_4
q_4	q_3	q_2

- q_1 is the start state
- $F = \{q_4\}$

There are infinitely many other correct DFAs for C.

(b) A regular expression for C is $(a \cup b)^* bab$. There are infinitely many other correct regular expressions for C.

4. A CFG for $D = \{\, c^i b^j c^k \mid i, j, k \geq 0,\ i = j + k \,\}$ is $G = (V, \Sigma, R, S)$ with set of variables $V = \{S, X\}$, where S is the start variable; set of terminals $\Sigma = \{b, c\}$; and rules

$$\begin{aligned} S &\to cSc \mid X \\ X &\to cXb \mid \varepsilon \end{aligned}$$

There are infinitely many other correct CFGs for D. For example, we could define R to instead be

$$\begin{aligned} S &\rightarrow cSc \mid X \mid \varepsilon \\ X &\rightarrow cXb \mid cb \mid \varepsilon \end{aligned}$$

5. Language $E = \{ w \in \Sigma^* \mid w = w^{\mathcal{R}} \text{ and } w \text{ has even length} \}$ with $\Sigma = \{0, 1\}$ is nonregular. We prove this by contradiction. Suppose that E is a regular language. Let p be the "pumping length" of the Pumping Lemma. Consider the string

$$s = a^p bba^p.$$

Note that $s \in E$ because $s^{\mathcal{R}} = s$ and its length $|s| = 2p + 2 = 2(p + 1)$ is even. Also, the length of s is $|s| = 2p + 2 > p$, so the Pumping Lemma will hold. Thus, there exists strings x, y, and z such that $s = xyz$ and

(i) $xy^iz \in E$ for each $i \geq 0$,

(ii) $|y| > 0$,

(iii) $|xy| \leq p$.

Since the first p symbols of s are all a's, the third property implies that x and y consist only of a's. So z will be the rest of the a's at the beginning, followed by bba^p. The second property states that $|y| > 0$, so y has at least one a. More precisely, we can then say that

$$\begin{aligned} x &= a^j \text{ for some } j \geq 0, \\ y &= a^k \text{ for some } k \geq 1, \\ z &= a^m bba^p \text{ for some } m \geq 0. \end{aligned}$$

Since $a^p bba^p = s = xyz = a^j a^k a^m bba^p = a^{j+k+m} bba^p$, we must have that

$$j + k + m = p, \text{ where } k \geq 1$$

by (ii). The first property implies that $xy^2z \in E$, but

$$\begin{aligned} xy^2z &= a^j a^k a^k a^m bba^p \\ &= a^{p+k} bba^p \notin E \end{aligned}$$

because $(a^{p+k} bba^p)^{\mathcal{R}} = a^p bba^{p+k}$ is not the same as $a^{p+k} bba^p$ since $k \geq 1$. Because the pumped string $xy^2z \notin E$, we have a contradiction. Therefore, E is a nonregular language.

A string that will not work for getting a contradiction is $s = 0^p \in E$, which has $|s| \geq p$, so the pumping lemma will apply. Then we could let $x = z = \varepsilon$ and $y = 0^p$, and every pumped string $xy^iz = 0^{ip} \in E$, so there is no contradiction. There are many other strings that won't work.

6. (This is HW 7, problem 2b.) We have to prove that the class of Turing-recognizable languages is closed under union. To do this, suppose that L_1 and L_2 are Turing-recognizable languages, and we need to show that their union $A_1 \cup A_2$ is also Turing-recognizable. Let M_1 and M_2 be TMs that recognize L_1 and L_2, respectively. We construct a TM M' that recognizes the union $L_1 \cup L_2$:

   M' = "On input string w:

   1. Run M_1 and M_2 alternately on w, one step at a time.
   If either accepts, *accept*. If both halt and reject, *reject*.

   To see why M' recognizes $L_1 \cup L_2$, first consider $w \in L_1 \cup L_2$. Then w is in L_1 or in L_2 (or both). If $w \in L_1$, then M_1 accepts w, so M' will eventually accept w. Similarly, if $w \in L_2$, then M_2 accepts w, so M' will eventually accept w. On the other hand, if $w \notin L_1 \cup L_2$, then $w \notin L_1$ and $w \notin L_2$. Thus, neither M_1 nor M_2 accepts w, so M' will also not accept w. Hence, M' recognizes $L_1 \cup L_2$. Note that if neither M_1 nor M_2 accepts w and one of them does so by looping, then M' will loop, but this is fine because we only needed M' to *recognize* and not *decide* $L_1 \cup L_2$.

7. $q_1bbab\#abb$ $\quad xq_3bab\#abb$ $\quad xbq_3ab\#abb$ $\quad xbaq_3b\#abb$ $\quad xbabq_3\#abb$ $\quad xbab\#q_5abb$ $\quad xbab\#aq_{\text{reject}}bb$

8. Multiple answers

   (a) For the given relations, the following are true:

   - F is a subset of D
   - N = R
   - P is a subset of T
   - N is a subset of G

   The rest are not true.

   (b) The given PDA recognizes the language $A = \{\, w \in \{0,1\}^* \mid w = w^{\mathcal{R}}, |w| \text{ is odd} \,\}$; see HW 6, problem 1b. Two of the given CFGs will generate A: rules

   $$S \rightarrow 0S0 \mid 1S1 \mid 0 \mid 1$$

   and rules

   $$\begin{aligned} S &\rightarrow 0S0 \mid X \mid Y \\ X &\rightarrow 1X1 \mid S \\ Y &\rightarrow 0 \mid 1 \end{aligned}$$

   None of the other CFGs are correct.

   (c) The class of CFLs is closed under union, concatenation, and Kleene star, but not under intersection and complements.

(d) The class of finite languages is closed under union, intersection, and concatenation. To see why the class is not closed under complementation, the finite language $A = \{\varepsilon, a, b\}$ with alphabet $\Sigma = \{a, b\}$ has complement $\overline{A} = \{ w \in \Sigma^* \mid |w| \geq 2\}$, which is infinite. Similarly, to see why the class is not closed under Kleene star, the same finite A has $A^* = \Sigma^*$, which is infinite.

(e) Language A is context-free, so there is a PDA, Turing machine, k-tape Turing machine and nondeterministic Turing machine that will recognize A.