CS 341-006, Spring 2022
Solutions for Midterm 1, Hybrid

1. (a) True. HW 4, problem 5(a).

   (b) True. HW 4, problem 5(c).

   (c) True. Suppose A is non-context-free but regular. But then Corollary 2.32 implies A is context-free, which is a contradiction.

   (d) True. If $A \subseteq B$, then $x \in A$ implies $x \in B$, so there is no $x \in A$ with $x \notin B$. Thus, $A \cap \overline{B} = \emptyset$.

   (e) False. For example, $A = \{0^n 1^n 0^n \mid n \geq 0\}$ is a subset of $B = L((0 \cup 1)^*)$, but A is non-context-free and B is context-free.

   (f) False. The language $\{\, a^n b^n \mid 5 \leq n \leq 20\} = \{a^5 b^5, a^6 b^6, \ldots, a^{20} b^{20}\}$ is finite. Thus, slide 1-95 implies the language is regular.

   (g) True. Because A has a regular expression, A is a regular language by Kleene's Theorem (1.54). Then Corollary 2.32 implies A is also context-free, so it has a CFG. Theorem 2.9 then ensures that A has a CFG in Chomsky normal form.

   (h) True. See slide 2-111.

   (i) True. By HW 2, problem 3, we know that $\overline{A}$ is regular. Because $\overline{A}$ and B are regular, then $\overline{A} \cup B$ is regular by Theorem 1.25. Theorem 1.49 then implies $(\overline{A} \cup B)^*$ is regular.

   (j) False. See HW 6, problem 2(a).

2. (a) $(a^*ba^*ba^*)^*a^*ba^*$. Another regular expression is $(a^*ba^*b)^*a^*ba^*$. There are infinitely many correct regular expressions for the language. But the regular expression $(a^*ba^*ba^*)^*ba^*$ is wrong because it cannot generate the string $ab \in A$.

   (b) $G_2 = (V_2, \Sigma, R_2, S_2)$, where

   - $V_2 = V_1 \cup \{S_2\}$,
   - S_2 is the starting variable, where $S_2 \notin V_1$
   - Σ is the same alphabet of terminals as in G_1, and
   - $R_2 = R_1 \cup \{\, S_2 \to S_1 S_2 \mid \varepsilon \,\}$.

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

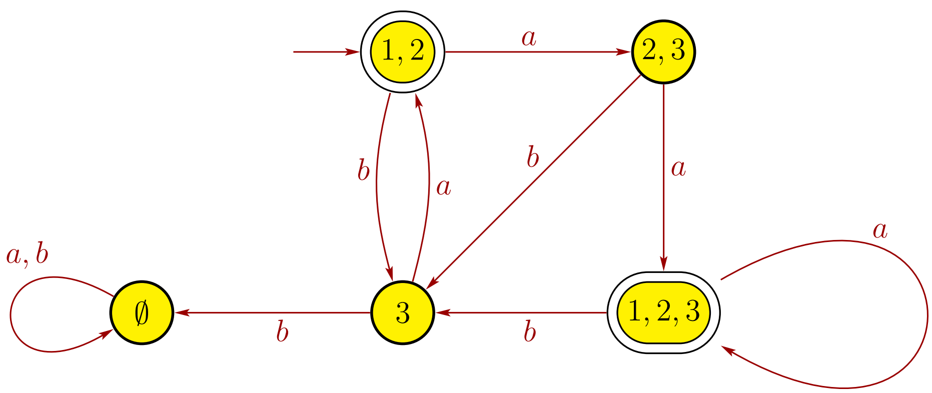
   - $Q_3 = Q_1 \times Q_2$;
   - Σ is the same alphabet as M_1 and M_2 have;
   - the transition function δ_3 satisfies $\delta_3((q, r), \ell) = (\delta_1(q, \ell), \delta_2(r, \ell))$ for $(q, r) \in Q_3$ and $\ell \in \Sigma$;
   - the starting state $q_3 = (q_1, q_2)$; and
   - $F_3 = (Q_1 \times F_2)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

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

$$\begin{aligned} S_0 &\to S \\ S &\to 0SA0SA \mid 0S0SA \mid 0SA0S \mid 0S0S \mid 0A0S \mid 00S \mid \varepsilon \\ A &\to 10A01 \mid 1001 \end{aligned}$$

3. (a) $\varepsilon, aa, ba, aaa, aba$

(b) A DFA for C is below:



Although the problem did not ask for it, a 5-tuple description of the DFA above is $M = (Q, \Sigma, \delta, \{1,2\}, F)$, where

- $Q = \{ \{1,2\}, \{2,3\}, \{3\}, \{1,2,3\}, \emptyset \}$
- $\Sigma = \{a, b\}$
- The transition function $\delta : Q \times \Sigma \to Q$ is defined as

	a	b
$\{1,2\}$	$\{2,3\}$	$\{3\}$
$\{2,3\}$	$\{1,2,3\}$	$\{3\}$
$\{3\}$	$\{1,2\}$	$\emptyset$
$\{1,2,3\}$	$\{1,2,3\}$	$\{3\}$
$\emptyset$	$\emptyset$	$\emptyset$

- $\{1,2\}$ is the start state
- $F = \{ \{1,2\}, \{1,2,3\} \}$

There are infinitely many other correct DFAs for A.

4. (a) This is a slight variation of HW 5, problem 1(f). A CFG for the language $D = \{ c^i a^j b^k \mid i, j, k \geq 0, \text{ and } j = i + k \}$ is $G = (V, \Sigma, R, S)$ with $V = \{S, X, Y\}$ as the set of variables, where S is the start variable; $\Sigma = \{a, b, c\}$ is the set of terminals; and rules R given by

$$\begin{aligned} S &\to XY \\ X &\to cXa \mid \varepsilon \\ Y &\to aYb \mid \varepsilon \end{aligned}$$

There are infinitely many other correct CFGs for D.

We next prove the correctness of the CFG G, although the problem doesn't requiring providing such a proof. To see why the given CFG G works for D, we first claim that $D = A_1 \circ A_2$, where

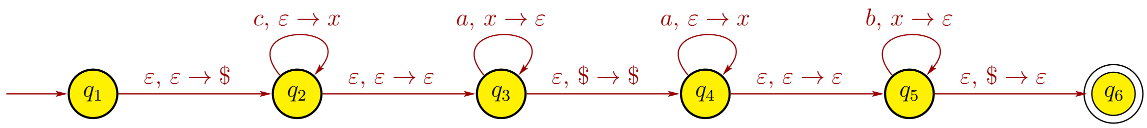
$$A_1 = \{ c^i a^i \mid i \geq 0 \},$$
$$A_2 = \{ a^k b^k \mid k \geq 0 \}.$$

To prove that $D = A_1 \circ A_2$, we need to show that both $A_1 \circ A_2 \subseteq D$ and $D \subseteq A_1 \circ A_2$.

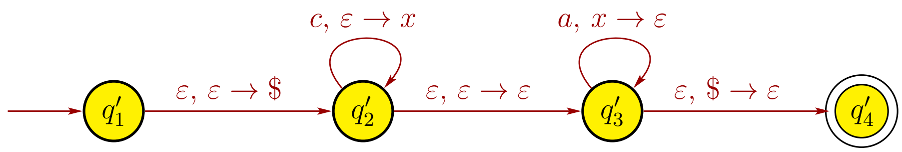
- To prove that $A_1 \circ A_2 \subseteq D$, we have to show that concatenating a string from A_1 with a string from A_2 always results in a string in D. This is true because concatenating a string $c^i a^i \in A_1$ with a string $a^k b^k \in A_2$ leads to $c^i a^i a^k b^k = c^i a^{i+k} b^k \in D$.
- Conversely, to show that $D \subseteq A_1 \circ A_2$, we need to show that every string in D can be expressed as a concatenation of a string from A_1 with a string from A_2. This is true because any string $s = c^i a^j b^k \in D$ has $j = i + k$, so $s = c^i a^{i+k} b^k = c^i a^i a^k b^k \in A_1 \circ A_2$.

In our CFG G, the rules $X \to cXa \mid \varepsilon$ with X as the starting variable result in the language A_1. The rules $Y \to aYb \mid \varepsilon$ with starting variable Y result in the language A_2. As shown in HW 5, problem 3(b), the class of context-free languages is closed under concatenation, and the approach in that problem leads to the given CFG G for D.

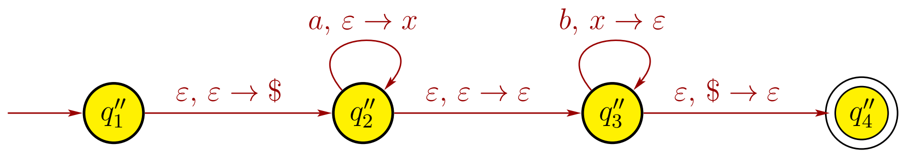
(b) This is a slight variation of HW 6, problem 1(g). A PDA M for D is as follows:



To understand the PDA M for $D = \{ c^i \, a^j \, b^k \mid i, j, k \geq 0 \text{ and } j = i + k \}$, the previous part explains that $D = L_1 \circ L_2$ because concatenating any string $c^i a^i \in L_1$ for $i \geq 0$ with any string $a^k b^k \in L_2$ for $k \geq 0$ results in a string $c^i a^i a^k b^k = c^i a^{i+k} b^k \in D$. Thus, for a string $c^i a^j b^k \in D$, which must have $j = i + k$, the number i of c's at the beginning has to be no more than the number j of a's in the middle (because $i + k = j$ implies $i \leq j$ since $i, j, k \geq 0$), and the remaining number $j - i$ of a's in the middle must match the number k of b's at the end. Hence, if we have PDAs M_1 and M_2 for L_1 and L_2, respectively, then we can then build a PDA for D by connecting M_1 and M_2 so that M_1 processes the first part of the string $c^i a^i$, and M_2 processes the second part of the string $a^k b^k$. A PDA M_1 for L_1 is



(We can get another PDA for L_1 by slightly modifying the one on slide 2-38 of the notes.) Similarly, a PDA M_2 for L_2 is



But in connecting the two PDAs M_1 and M_2 to get a PDA M for D, we need to make sure the stack is empty after M_1 finishes processing the first part of the string and before M_2 starts processing the second part of the string. This is accomplished in the PDA M for D by the transition from q_3 to q_4 with label "$\varepsilon, \$ \to \$$".

There are infinitely many other correct PDAs for D.

There are also infinitely many incorrect PDAs for D. For example, in the given solution, if we change the label on the transition from q_3 to q_4 to instead be "$\varepsilon, \varepsilon \to \varepsilon$", then the PDA would incorrectly accept the string $cabb \not\in D$ by not looping in q_3 but instead looping once in q_4.

5. Language $D = \{ c^i a^j b^k \mid i, j, k \geq 0 \text{ and } j = i + k \}$ is nonregular. We prove this by contradiction. Suppose that D is a regular language. Let p be the "pumping length" of the Pumping Lemma. Consider the string $s = c^p a^{2p} b^p$. (Other possible strings that can work in the proof (with appropriate modifications) are $a^p b^p$ or $c^p a^p$.) Note that $s \in D$ because $s \in L(c^* a^* b^*)$, with the sum of the numbers of c's and b's in s equaling the number of a's in the middle. Also, $|s| = 3p > p$, so all of the assumptions of the Pumping Lemma hold. Thus, there exists strings x, y, and z such that $s = xyz$ and

(i) $xy^i z \in D$ for each $i \geq 0$,

(ii) $|y| > 0$,

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

Since the first p symbols of s are all c's, the third property implies that x and y consist only of c's. So z will be the rest of the c's, followed by $a^{2p} b^p$. The second property states that $|y| > 0$, so y has at least one c. More precisely, we can then say that

$$\begin{aligned} x &= c^j \text{ for some } j \geq 0, \\ y &= c^k \text{ for some } k \geq 1, \\ z &= c^m a^{2p} b^p \text{ for some } m \geq 0. \end{aligned}$$

Since $c^p a^{2p} b^p = s = xyz = c^j c^k c^m a^{2p} b^p = c^{j+k+m} a^{2p} b^p$, we must have that

$$j + k + m = p \quad \text{and} \quad k \geq 1.$$

The first property implies that $xy^2z \in D$, but

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

since $p + k + p > 2p$ because $j + k + m = p$ and $k \geq 1$, so in the pumped string xy^2z, the sum of the numbers of c's and b's doesn't match the number of a's. Because the pumped string $xy^2z \notin D$, we have a contradiction. Therefore, D is a nonregular language.