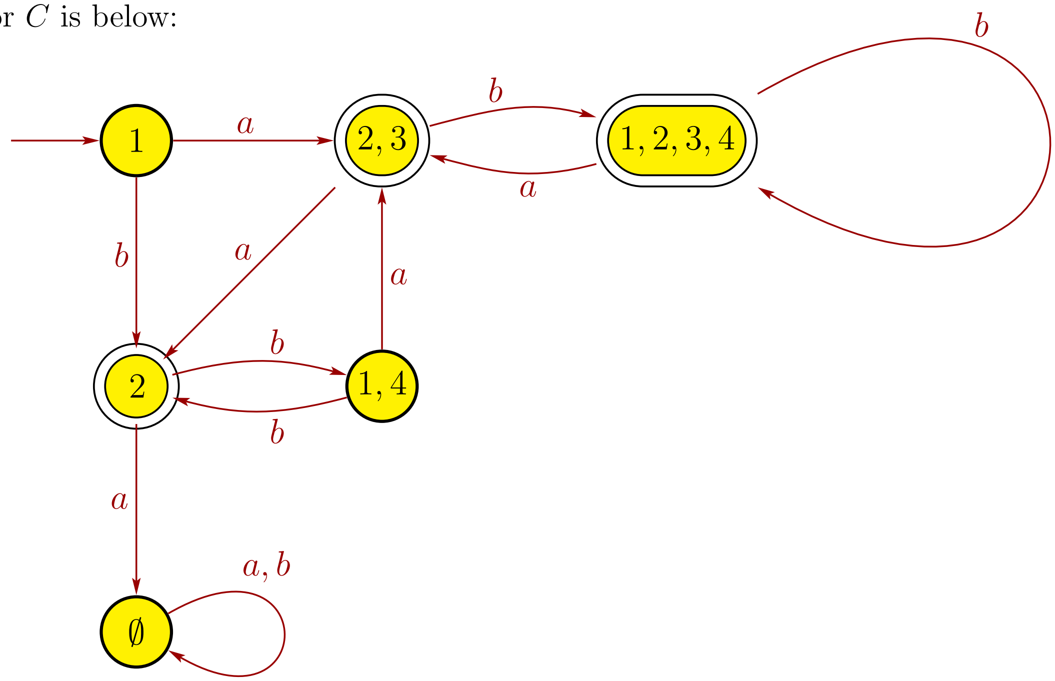
CS 341, Fall 2018, Face-to-Face Section
Solutions for Midterm 1

1. (a) False. Let $A = \emptyset$ and $B = \{ a^n b^n \mid n \geq 0 \}$. Then $A \subseteq B$, A is regular because it's finite, and B is nonregular.
   (b) False. The language a^* is regular but infinite.
   (c) False. $A = \{ a^n b^n \mid n \geq 0 \}$ is context-free but not regular.
   (d) True. Homework 2, problem 5.
   (e) False. $0^*1^*0^*$ generates the string $001000 \notin A$, so the regular expression is not correct. In fact, A is nonregular, so it can't have a regular expression.
   (f) True. Theorem 1.54 implies A is regular. Then by Corollary 2.32, A is context-free, so Theorem 2.20 ensures that A has a PDA.
   (g) True, by Lemma 2.27 and Theorem 2.9.
   (h) False. The transition function of an NFA is $\delta : Q \times \Sigma_\varepsilon \to \mathcal{P}(Q)$.
   (i) False. If A is nonregular, then A cannot have an NFA by Corollary 1.40.
   (j) False. Let $A = \{ a^n b^n \mid n \geq 0 \}$ and $B = (a \cup b)^*$. Then $A \subseteq B$, A is nonregular, and B is regular.

2. (a) $b^*ab^* \cup b^*aa^*bb^*$. Another regular expression is $b^*(a \cup aa^*b)b^*$. There are infinitely many correct regular expressions for the language.
   (b) $G' = (V', \Sigma, R', S_0)$ with $S_0 \notin V$, where
      - $V' = V \cup \{S_0\}$,
      - S_0 is the (new) starting variable,
      - Σ is the same alphabet of terminals as in G, and
      - $R' = R \cup \{S_0 \to SS_0 \mid \varepsilon\}$.
   (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

$$\begin{aligned} S_0 &\to S \\ S &\to A1AS \mid 1AS \mid A1S \mid 1S \mid 01A \mid 01 \mid \varepsilon \\ A &\to 1S0 \end{aligned}$$

3. A DFA for C is below:

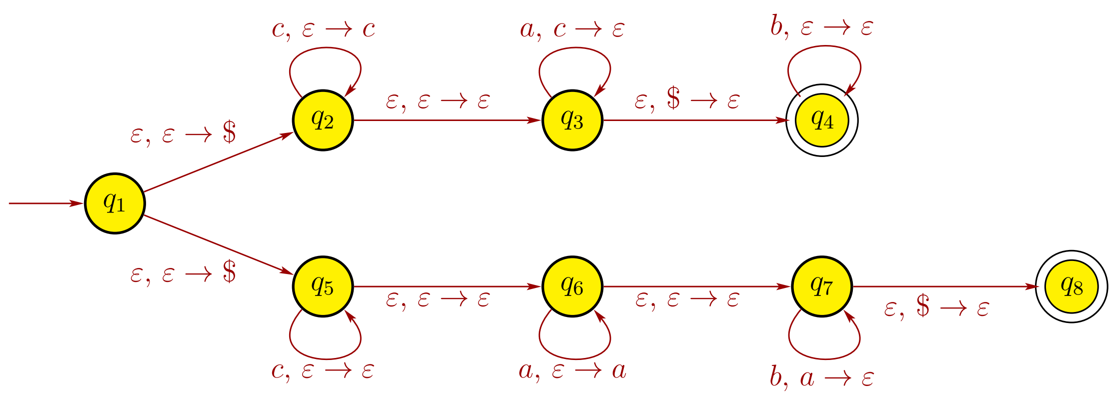


4. (a) $G = (V, \Sigma, R, S)$ with set of variables $V = \{S, W, X, Y, Z\}$, where S is the start variable; set of terminals $\Sigma = \{a, b, c\}$; and rules

$$\begin{aligned} S &\rightarrow WX \mid YZ \\ W &\rightarrow cWa \mid \varepsilon \\ X &\rightarrow bX \mid \varepsilon \\ Y &\rightarrow cY \mid \varepsilon \\ Z &\rightarrow aZb \mid \varepsilon \end{aligned}$$

Starting from variable W, the derived string will be in $A_1 = \{\, c^n a^n \mid n \geq 0 \,\}$. Starting from variable X, the derived string will be in $A_2 = L(b^*)$. So if $S \Rightarrow WX$ is the first step taken in a derivation, then the resulting string will be in the language $B_1 = A_1 \circ A_2 = \{\, c^n a^n b^k \mid n \geq 0, k \geq 0 \,\}$. A similar argument will show that if $S \Rightarrow YZ$ is the first step taken in a derivation, then the resulting string will be in the language $B_2 = \{\, c^i a^n b^n \mid i \geq 0, n \geq 0\}$, and note that $L = B_1 \cup B_2$. There are infinitely many other correct CFGs for L.

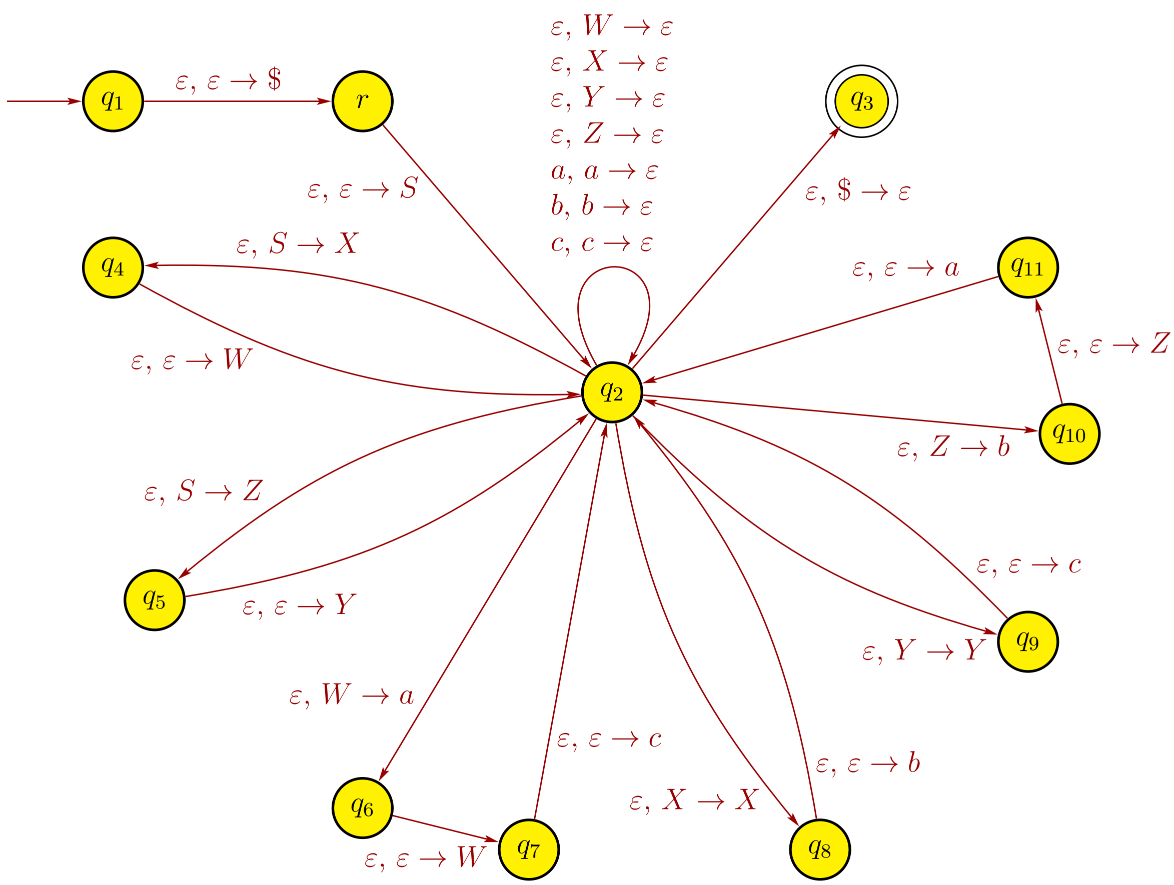
(b) There are infinitely many correct PDAs for L. Here is one:



The PDA has a nondeterministic branch at q_1.

- If the string is $c^i a^j b^k$ with $i = j$, then the PDA can accept the string by taking the branch from q_1 to q_2.
- If the string is $c^i a^j b^k$ with $j = k$, then the PDA can accept the string by taking the branch from q_1 to q_5.

Yet another approach uses the algorithm from Lemma 2.21 to convert the CFG in part (a) into a PDA.



Note that

- The path $q_2 \to q_4 \to q_2$ corresponds to the rule $S \to WX$.
- The path $q_2 \to q_5 \to q_2$ corresponds to the rule $S \to YZ$.
- The path $q_2 \to q_6 \to q_7 \to q_2$ corresponds to the rule $W \to cWa$.
- The path $q_2 \to q_8 \to q_2$ corresponds to the rule $X \to bX$.
- The path $q_2 \to q_9 \to q_2$ corresponds to the rule $Y \to cY$.
- The path $q_2 \to q_{10} \to q_{11} \to q_2$ corresponds to the rule $Z \to aZb$.

5. Language A is nonregular. We prove this by contradiction. Suppose that A is a regular language. Let p be the "pumping length" of the Pumping Lemma. Consider the string $s = c^p a^p$. Note that $s \in A$ because the numbers of c's and a's are equal, and $|s| = 2p > p$, so the Pumping Lemma will hold. Thus, there exists strings x, y, and z such that $s = xyz$ and

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

   (b) $|y| > 0$,

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

   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^p. The second property states that $|y| > 0$, so y has at least one c. More precisely, we can then say that

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

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

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

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

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

   since $p + k > p$ because $j + k + m = p$ and $k \geq 1$, so the number of c's in the pumped string $xy^2 z$ doesn't match the number of a's, and the number of a's doesn't match the number of b's (none). Because the pumped string $xy^2 z \notin A$, we have a contradiction. Therefore, A is a nonregular language.

   Note that if you instead chose the string $s = c^p a^p b^p$, you would not get a contradiction. This is because pumping up or down leads to the number of c's changing, but the number of a's and b's remain the same and equal. Thus, the pumped string is still in the language, so there is no contradiction.