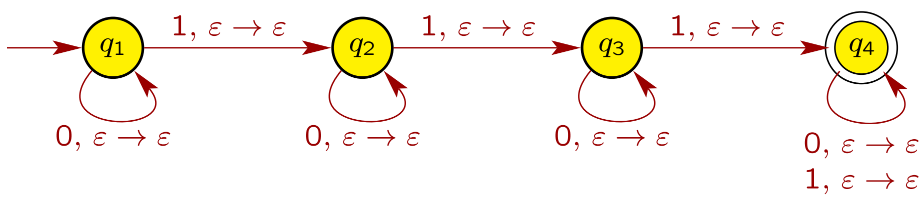
CS 341: Foundations of Computer Science II
Prof. Marvin Nakayama

Homework 6 Solutions

1. Give pushdown automata that recognize the following languages.

   (a) $A = \{ w \in \{0,1\}^* \mid w \text{ contains at least three } 1\text{s} \}$

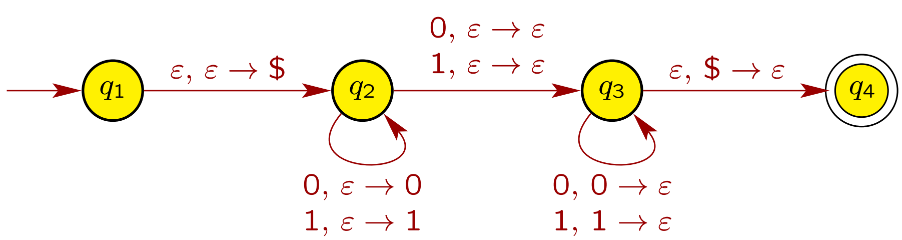
   Answer:



   Note that A is a regular language, so the language has a DFA. We can easily convert the DFA into a PDA by using the same states and transitions and never push nor pop anything from the stack.

   (b) $B = \{ w \in \{0,1\}^* \mid w = w^{\mathcal{R}} \text{ and the length of } w \text{ is odd} \}$

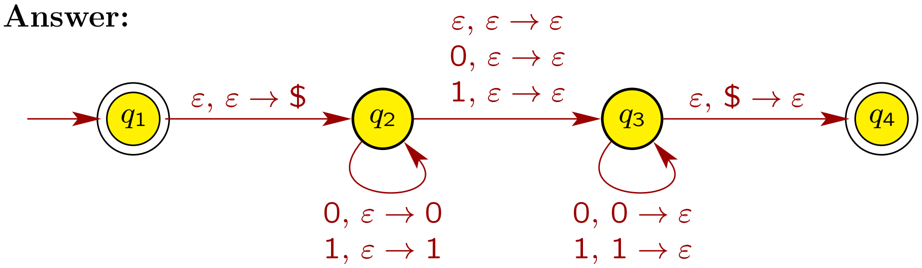
   Answer:



   Since the length of any string $w \in B$ is odd, w must have a symbol exactly in the middle position; i.e., $|w| = 2n + 1$ for some $n \geq 0$, and the $(n+1)$th symbol in w is the middle one. If a string w of length $2n + 1$ satisfies $w = w^{\mathcal{R}}$, the first n symbols must match (in reverse order) the last n symbols, and the middle symbol doesn't have to match anything. Thus, in the above PDA, the transition from q_2 to itself reads the first n symbols and pushes these on the stack. The transition from q_2 to q_3 nondeterministically identifies the middle symbol of w, which doesn't need to match any symbol, so the stack is unaltered. The transition from q_3 to itself then reads the last n symbols of w, popping the stack at each step to make sure the symbols after the middle match (in reverse order) the symbols before the middle.

(c) $C = \{ w \in \{0, 1\}^* \mid w = w^{\mathcal{R}} \}$

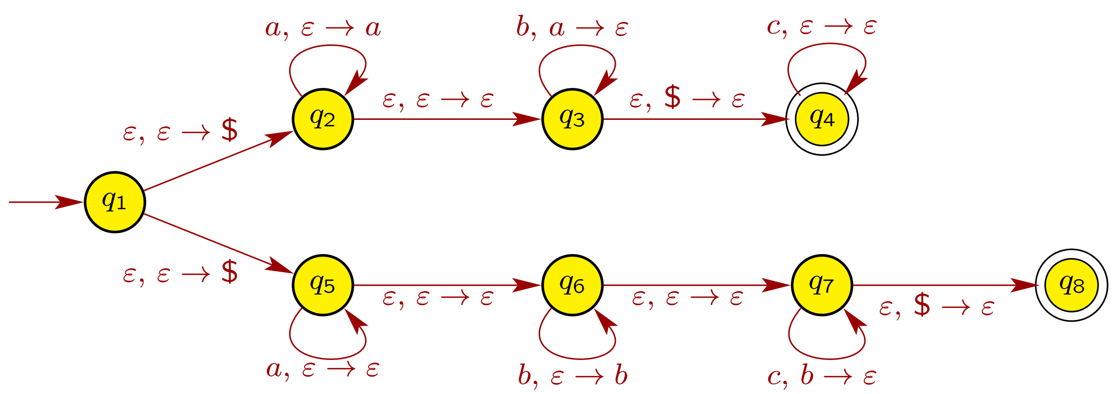
Answer:



The length of a string $w \in C$ can be either even or odd. If it's even, then there is no middle symbol in w, so the first half of w is pushed on the stack, we move from q_2 to q_3 without reading, pushing, or popping anything, and then match the second half of w to the first half in reverse order by popping the stack. If the length of w is odd, then there is a middle symbol in w, and the description of the PDA in part (b) applies.

(d) $D = \{ a^i\, b^j\, c^k \mid i, j, k \geq 0, \text{ and } i = j \text{ or } j = k \}$

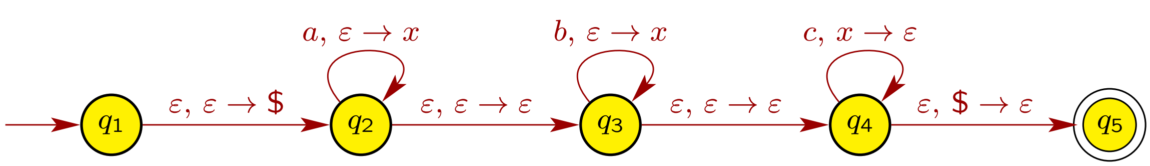
Answer:



The PDA has a nondeterministic branch at q_1. If the string is $a^i b^j c^k$ with $i = j$, then the PDA takes the branch from q_1 to q_2. If the string is $a^i b^j c^k$ with $j = k$, then the PDA takes the branch from q_1 to q_5.

(e) $E = \{ a^i\, b^j\, c^k \mid i, j, k \geq 0 \text{ and } i + j = k \}$

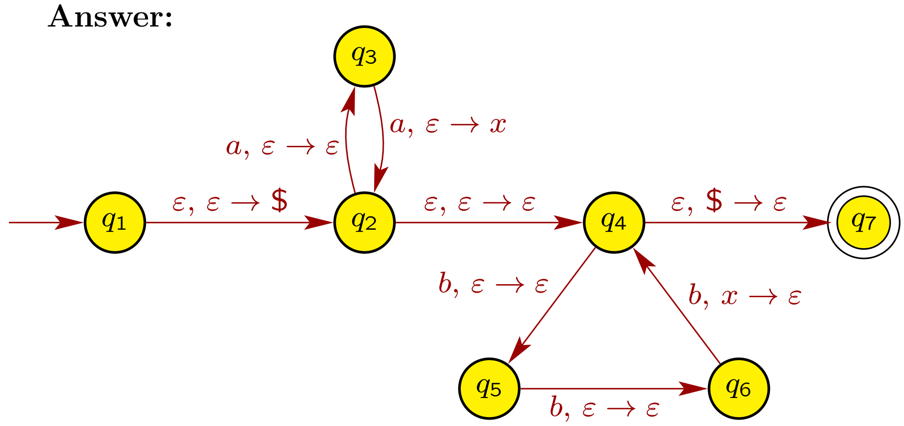
Answer:



For every a and b read in the first part of the string, the PDA pushes an x onto the stack. Then it must read a c for each x popped off the stack.

(f) $F = \{ a^{2n}b^{3n} \mid n \geq 0 \}$

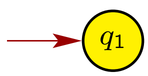
Answer:



The PDA pushes a single x onto the stack for every 2 a's read at the beginning of the string. Then it pops a single x for every 3 b's read at the end of the string.

(g) $\emptyset$

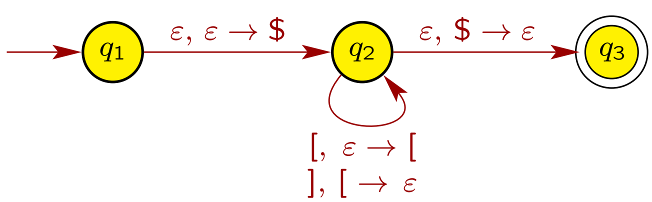
Answer:



Since the PDA has no accept states, the PDA accepts no strings; i.e., the PDA recognizes the language $\emptyset$.

(h) The language H of strings of properly balanced left and right brackets: every left bracket can be paired with a unique subsequent right bracket, and every right bracket can be paired with a unique preceding left bracket. Moreover, the string between any such pair has the same property. For example, $[\,]\,[\,[\,[\,]\,[\,]\,]\,[\,]\,] \in A$.

Answer:



2. (a) Use the languages

$$\begin{aligned} A &= \{ a^m b^n c^n \mid m, n \geq 0 \} \quad \text{and} \\ B &= \{ a^n b^n c^m \mid m, n \geq 0 \} \end{aligned}$$

together with Example 2.36 of the textbook to show that the class of context-free languages is not closed under intersection.

Answer: The language A is context free since it has CFG G_1 with rules

$$\begin{aligned} S &\rightarrow XY \\ X &\rightarrow aX \mid \varepsilon \\ Y &\rightarrow bYc \mid \varepsilon \end{aligned}$$

The language B is context free since it has CFG G_2 with rules

$$\begin{aligned} S &\rightarrow XY \\ X &\rightarrow aXb \mid \varepsilon \\ Y &\rightarrow cY \mid \varepsilon \end{aligned}$$

But $A \cap B = \{ a^n b^n c^n \mid n \geq 0 \}$, which we know is not context free from Example 2.36 of the textbook. Thus, the class of context-free languages is not closed under intersection.

(b) Use part (a) and DeMorgan's law (Theorem 0.20 of the textbook) to show that the class of context-free languages is not closed under complementation.

Answer: We will use a proof by contradiction, so we first assume the opposite of what we want to show; i.e., suppose the following is true:

R1. The class of context-free languages is closed under complementation.

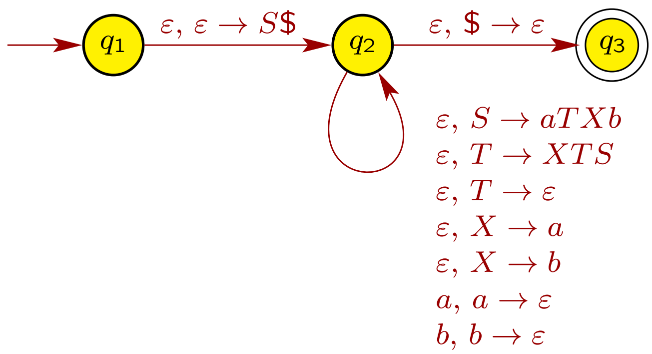
Define the context-free languages A and B as in the previous part. Then R1 implies $\overline{A}$ and $\overline{B}$ are context-free. We know the class of context-free languages is closed under union, as shown on slide 2-101, so we then must have that $\overline{A} \cup \overline{B}$ is context-free. Then again apply R1 to conclude that $\overline{\overline{A} \cup \overline{B}}$ is context-free. Now DeMorgan's law states that $A \cap B = \overline{\overline{A} \cup \overline{B}}$, but we showed in the previous part that $A \cap B$ is not context-free, which is a contradiction. Therefore, R1 must not be true.

3. Consider the following CFG $G = (V, \Sigma, R, S)$, where $V = \{S, T, X\}$, $\Sigma = \{a, b\}$, the start variable is S, and the rules R are

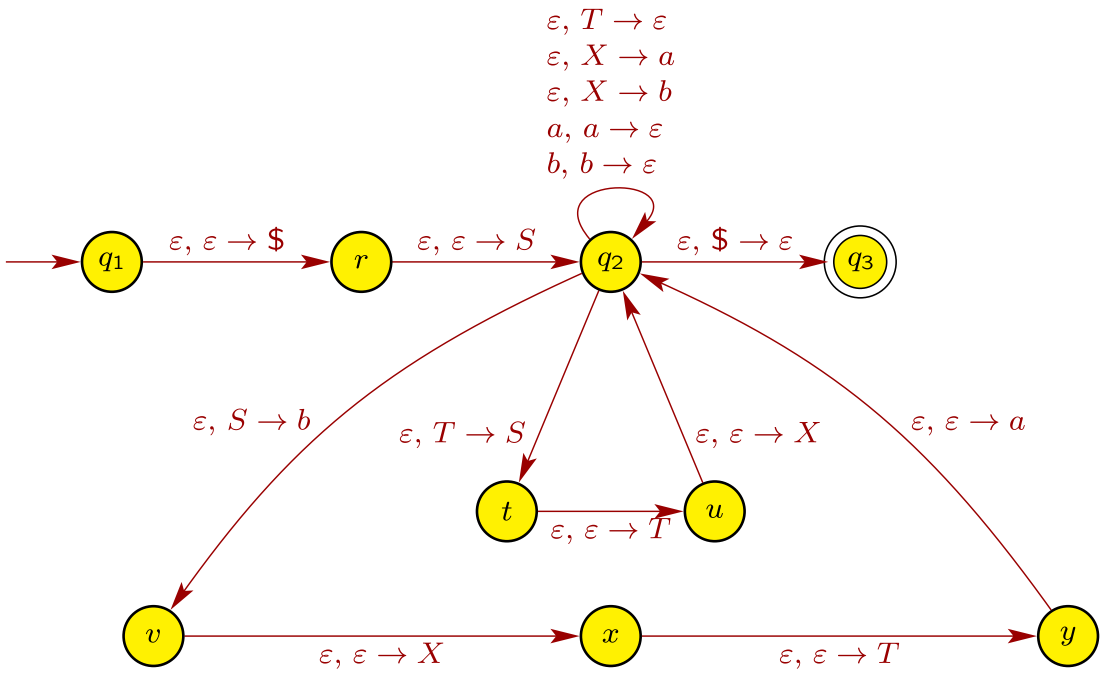
$$\begin{aligned} S &\rightarrow aTXb \\ T &\rightarrow XTS \mid \varepsilon \\ X &\rightarrow a \mid b \end{aligned}$$

Convert G to an equivalent PDA using the procedure given in Lemma 2.21.

Answer: First we create a PDA for G that allows for pushing strings onto the stack:



Then we need to fix the non-compliant transitions, i.e., the ones for which a string of length more than 1 is pushed onto the stack. The only non-compliant transitions are the first two from q_2 back to itself, and the transition from q_1 to q_2. Fixing these gives the following PDA:



4. Use the pumping lemma to prove that the language $A = \{ 0^{2n}\, 1^{3n}\, 0^n \mid n \geq 0 \}$ is not context free.

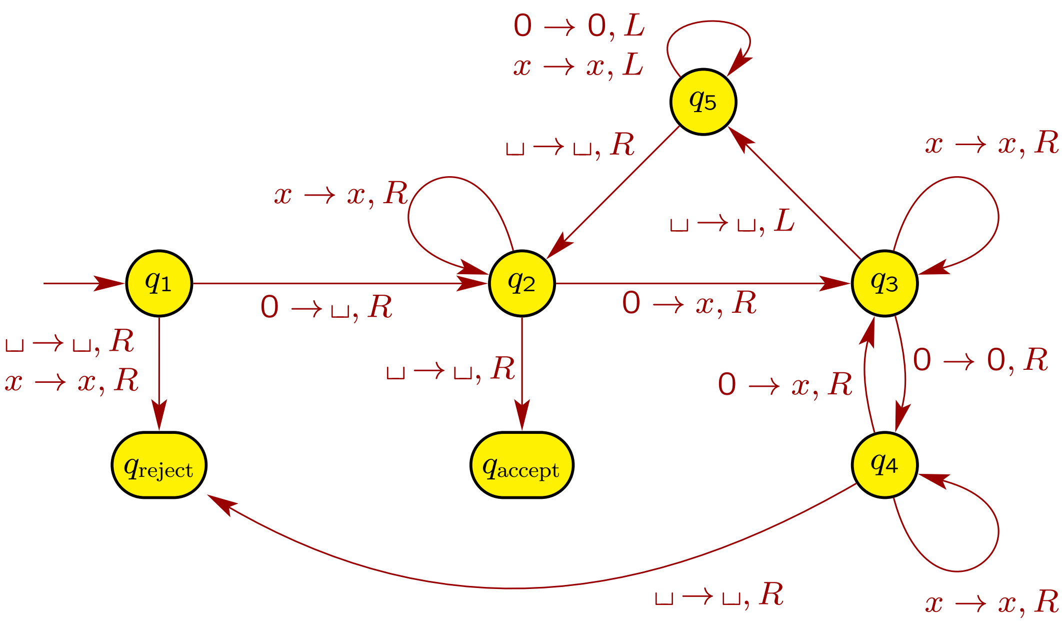
   Answer: Assume that A is a CFL. Let p be the pumping length of the pumping lemma for CFLs, and consider string $s = 0^{2p}1^{3p}0^p \in A$. Note that $|s| = 6p > p$, so the pumping lemma will hold. Thus, there exist strings u, v, x, y, z such that $s = uvxyz = 0^{2p}1^{3p}0^p$, $uv^ixy^iz \in A$ for all $i \geq 0$, and $|vy| \geq 1$. We now consider all of the possible choices for v and y:

   - Suppose strings v and y are uniform (e.g., $v = 0^j$ for some $j \geq 0$, and $y = 1^k$ for some $k \geq 0$). Then $|vy| \geq 1$ implies that $j \geq 1$ or $k \geq 1$ (or both), so uv^2xy^2z won't have the correct number of 0's at the beginning, 1's in the middle, and 0's at the end. Hence, $uv^2xy^2z \notin A$.

- Now suppose strings v and y are not both uniform. Then uv^2xy^2z will not have the form $0 \cdots 01 \cdots 10 \cdots 0$. Hence, $uv^2xy^2z \notin A$.

Thus, there are no options for v and y such that $uv^ixy^iz \in A$ for all $i \geq 0$. This is a contradiction, so A is not a CFL.

5. The Turing machine M below recognizes the language $A = \{ 0^{2^n} \mid n \geq 0 \}$.



In each of the parts below, give the sequence of configurations that M enters when started on the indicated input string.

(a) 00

Answer: $q_1 00$ $\quad \sqcup q_2 0$ $\quad \sqcup x q_3 \sqcup$ $\quad \sqcup q_5 x$ $\quad q_5 \sqcup x$ $\quad \sqcup q_2 x$ $\quad \sqcup x q_2 \sqcup$ $\quad \sqcup x \sqcup q_{\text{accept}}$

(b) 000000

Answer: $q_1 000000$ $\quad \sqcup q_2 00000$ $\quad \sqcup x q_3 0000$ $\quad \sqcup x 0 q_4 000$
$\sqcup x 0 x q_3 00$ $\quad \sqcup x 0 x 0 q_4 0$ $\quad \sqcup x 0 x 0 x q_3 \sqcup$ $\quad \sqcup x 0 x 0 q_5 x$ $\quad \sqcup x 0 x q_5 0 x$
$\sqcup x 0 q_5 x 0 x$ $\quad \sqcup x q_5 0 x 0 x$ $\quad \sqcup q_5 x 0 x 0 x$ $\quad q_5 \sqcup x 0 x 0 x$ $\quad \sqcup q_2 x 0 x 0 x$
$\sqcup x q_2 0 x 0 x$ $\quad \sqcup x x q_3 x 0 x$ $\quad \sqcup x x x q_3 0 x$ $\quad \sqcup x x x 0 q_4 x$ $\quad \sqcup x x x 0 x q_4 \sqcup$
$\sqcup x x x 0 x \sqcup q_{\text{reject}}$