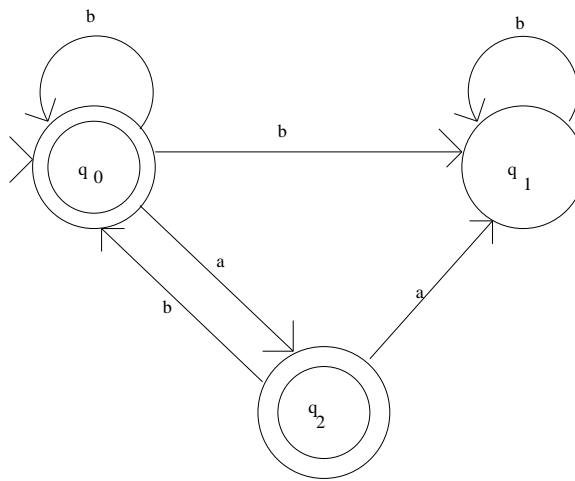


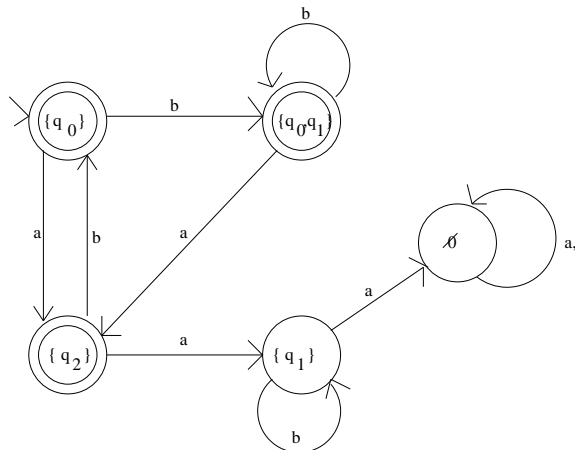
CS 3133 Foundations of Computer Science
C term 2018

Solutions for the Practice Final Exam

1. Construct the state diagram of a DFA equivalent to the following NFA



Solution:



2. Consider the following grammar G :

$$\begin{aligned} S &\rightarrow aSdd|A \\ A &\rightarrow bAc|bc \end{aligned}$$

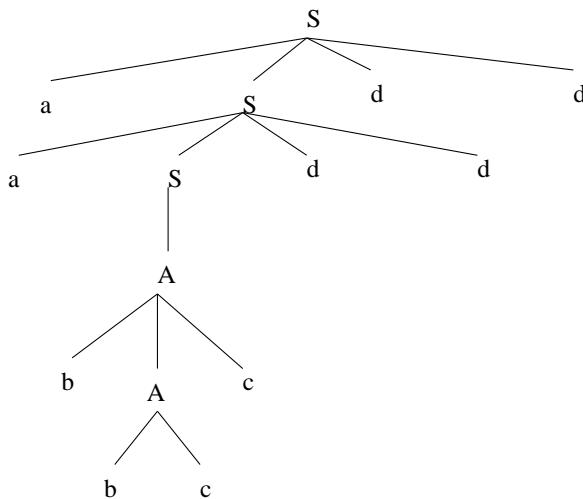
- (a) Give a leftmost derivation of $aabbccdddd$.
- (b) Build the derivation tree for the derivation in part (a).
- (c) What is $L(G)$?

Solution:

- (a) The following is a leftmost derivation of $aabbccdddd$:

$$\begin{aligned} S &\Rightarrow aSdd \\ &\Rightarrow aaSdddd \\ &\Rightarrow aaAdddd \\ &\Rightarrow aabAcdddd \\ &\Rightarrow aabbccdddd \end{aligned}$$

- (b) Here is the derivation tree:



- (c)

$$L(G) = \{a^n b^m c^m d^{2n} \mid n \geq 0, m > 0\}$$

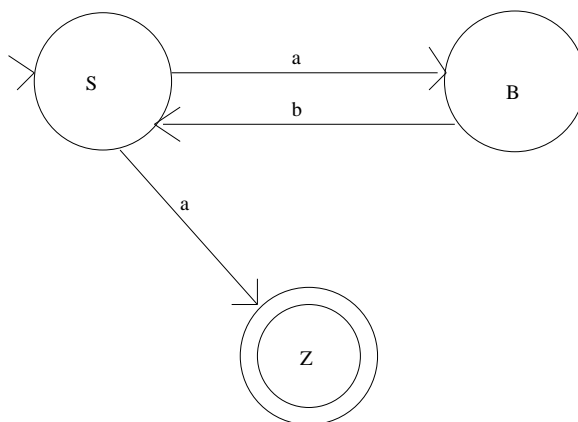
3. Consider the following regular grammar G :

$$\begin{aligned} S &\rightarrow aB|a \\ B &\rightarrow bS. \end{aligned}$$

- (a) Build an NFA M that accepts $L(G)$.
- (b) Construct a regular grammar G' from M that generates $L(M)$.
What is the difference between G and G' ?
- (c) Give a regular expression for $L(G)$.

Solution:

(a)



(b) The grammar G' is the following:

$$\begin{aligned} S &\rightarrow aB|aZ \\ B &\rightarrow bS \\ Z &\rightarrow \lambda \end{aligned}$$

Thus the only difference is that $S \rightarrow a$ is replaced by $S \rightarrow aZ$ and $Z \rightarrow \lambda$.

(c)

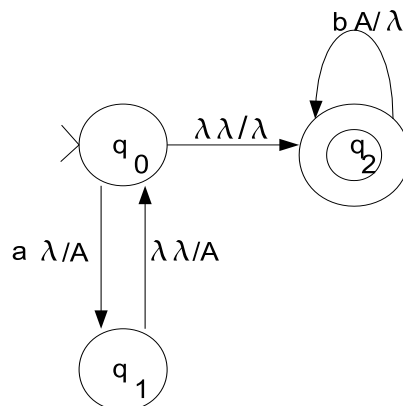
$$L(G) = (ab)^*a$$

4. Construct a PDA with $\Sigma = \{a, b\}$ that accepts the language

$$\{a^i b^{2i} | i \geq 0\}.$$

Is this language context-free? Is this language regular? Justify your answers!

Solution:



Of course it is context-free since it is accepted by a PDA. However, it is not regular as can be seen by applying the Pumping Lemma for regular languages. Let us assume indirectly that the language $L = \{a^i b^{2i} | i \geq 0\}$ is regular. This implies that L is accepted by some DFA. Let k be the number of states of the DFA. By the pumping lemma, every string $z \in L$ of length k or more can be decomposed into substrings u, v and w such that $\text{length}(uv) \leq k$, $\text{length}(v) > 0$ and $uv^i w \in L$ for all $i \geq 0$.

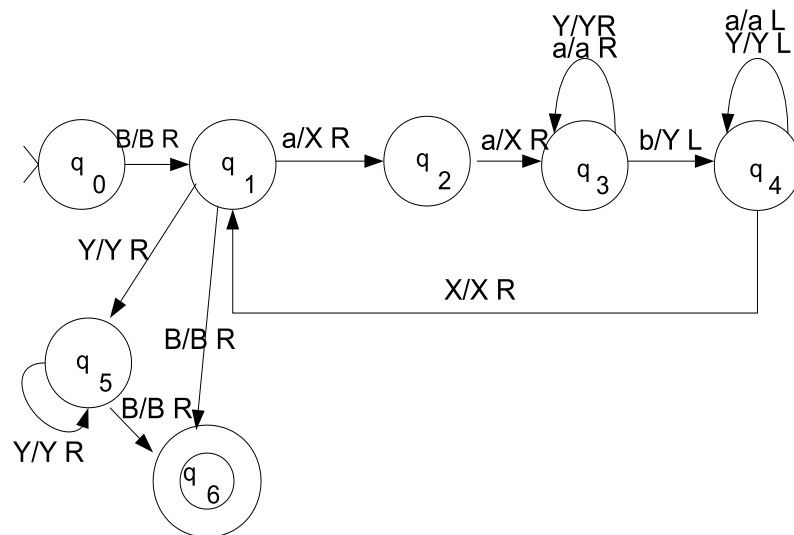
Consider the string $z = a^k b^{2k}$. Clearly $z \in L$ and $\text{length}(z) \geq k$. Using the pumping lemma we decompose z into substrings u, v and w , where $0 < \text{length}(uv) \leq k$. Then v is a substring of the a^k . But in this case $uv^2 w$ cannot be in L (since the number of a 's is more than half of the number of b 's), a contradiction. L is non-regular.

- Design a standard Turing machine with $\Sigma = \{a, b\}$ that accepts the language

$$\{a^{2i} b^i | i \geq 0\}.$$

Is this language recursive?

Solution:



Of course it is recursive since it is accepted by a Turing machine that halts on every input.