1. Extra Problems for Lecture 14

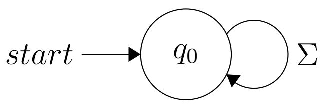
a. In the context of natural languages, what is the type (set, string, number, etc.) and meaning of each of these symbols?

$$\Sigma \qquad \Sigma^* \qquad \varepsilon$$

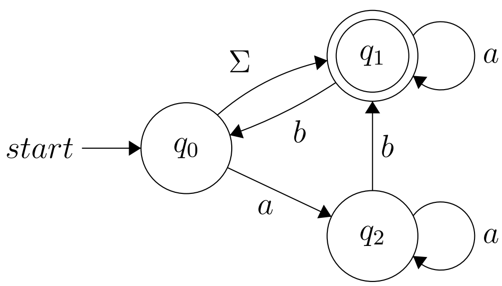
b. Let $\Sigma = \{a, b\}$. Are each of these automata DFAs over Σ? Why or why not? If it is a DFA, what language does it accept? (Hint: It's a good idea to test out some strings and see which ones work.)

Here are the rules for DFAs, as a refresher: For each state in the DFA, there must be exactly one transition defined for each symbol in Σ; there is a unique start state; and there are zero or more accepting states.

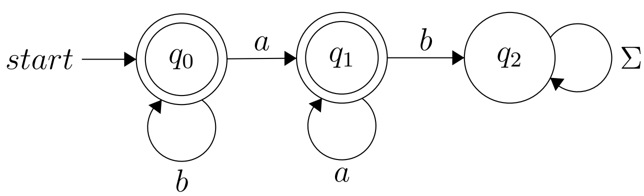
(1)



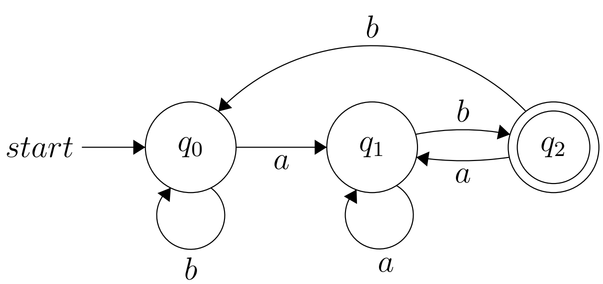
(2)



(3)



(4)



2. Extra Problems for Lecture 15

a. Let $\Sigma = \{n, a\}$. Design a DFA that recognizes each of these languages:

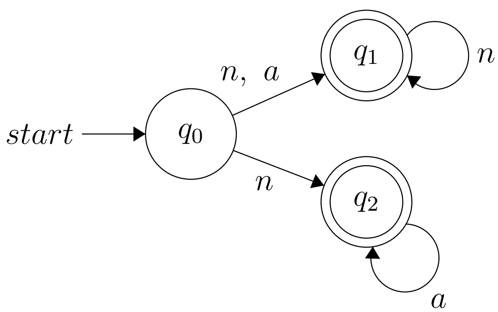
(1) {`ann`}

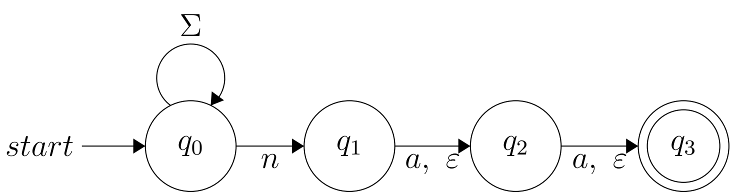
(2) $\bar{L}$, given that $L = \{\varepsilon\}$ (Recall that the notation $\bar{L}$ refers to the complement of the language L, meaning the language of all strings in Σ^* that are not in L.)

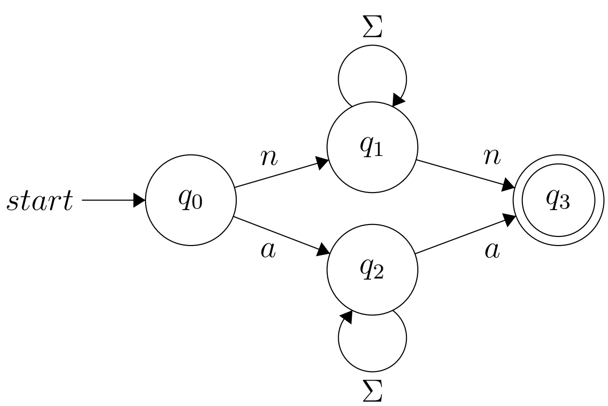
(3) $\{w \in \Sigma^* \mid w \text{ has an odd length }\}$

(4) $\{\varepsilon, \texttt{na}, \texttt{nana}, \texttt{nanana}, ...\}$

b. Let $\Sigma = \{n, a\}$. Given the following NFAs over Σ, determine which language the NFA recognizes. It's okay to describe the language with set-builder notation or with words. Recall that an NFA accepts if any set of choices results in the automaton ending up in an accept state with the entire input having been read through.

(1) 

(2) 

(3) 

3. Extra Problems for Lecture 16

a. Read the Guide to the Subset Construction on the website. Then, use the subset construction to convert the DFA from 2b(3) into an NFA.

b. What is the relationship between regular languages and NFAs/DFAs?

c. If L_1 and L_2 are languages over Σ, describe the language $L_1 - L_2$ in words. Is this language regular? Why or why not? (Hint: Can you express it in terms of language operations on languages you know are regular?)