CPSC 313 — Tutorial Exercise #2

Interpretation and Design of Deterministic Finite Automata

About This Exercise

The following lecture concerns material found in Section 1.1 of *Introduction to the Theory of Computation* and presented in the following lectures.

- Lecture #2: Introduction to Deterministic Finite Automata
- Lecture #3: Design of Deterministic Finite Automata

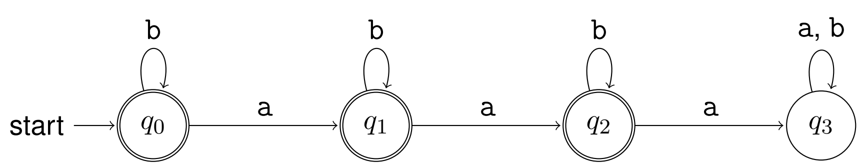
While questions on the "Interpretation of Deterministic Finite Automata", below, might be ***very*** briefly discussed, most of the tutorials should likely be spent on the (probably, more challenging) questions on "Design of Finite Automata".

This exercise will be discussed in tutorial on Thursday, January 20. Please try to solve the problems in this exercise ***before*** attending this tutorial, so that you can participate in discussions of them or joint efforts by students to solve them.

Problems To Be Solved

Interpretation of Deterministic Finite Automata

The following questions concern a deterministic finite automaton M that has alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ and that can be represented as follows.



1. Give the set Q of ***states*** in M and identify the ***start state***.
2. Give the set F of ***accepting states*** in M.
3. Describe the ***transition function*** $\delta : Q \times \Sigma \to Q$ by completing the following ***transition table***.

	a	b
q_0		
q_1		
q_2		
q_3		

4. Trace the execution of M on each of the following input strings — listing the sequence of states that are visited as symbols in the string are seen and processed, and stating whether the string is in the language of M.
   (a) λ
   (b) `a`
   (c) `b`
   (d) `ab`
   (e) `ba`
   (f) `abbab`
   (g) `aabbab`
   (h) `aaaabbb`
5. Give a ***brief*** description, in simple English, for each of the following subsets of $\Sigma^\star$.
   (a) $\{\omega \in \Sigma^\star \mid \delta^\star(q_0, \omega) = q_0\}$
   (b) $\{\omega \in \Sigma^\star \mid \delta^\star(q_0, \omega) = q_1\}$
   (c) $\{\omega \in \Sigma^\star \mid \delta^\star(q_0, \omega) = q_2\}$
   (d) $\{\omega \in \Sigma^\star \mid \delta^\star(q_0, \omega) = q_3\}$
6. Use your answer for the previous question to give a ***brief*** description, in simple English, of the language of (this particular DFA) M.

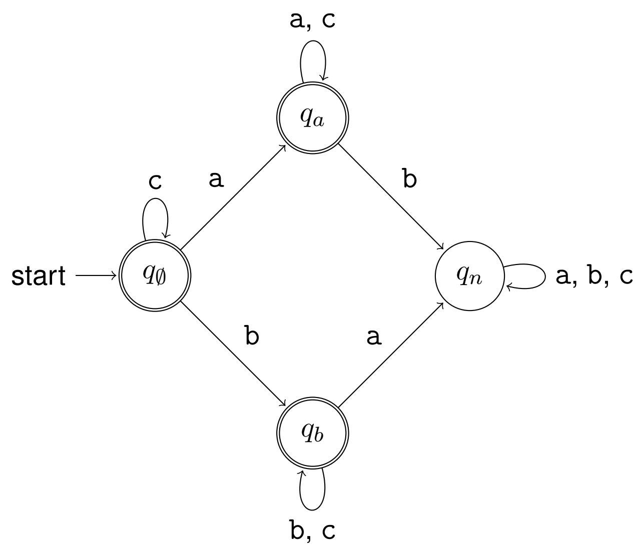


Figure 1: A Deterministic Finite Automaton

Design of Deterministic Finite Automata

6. Let $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$. Consider the deterministic finite automaton M for the language

   $$L_2 = \{\omega \in \Sigma^\star \mid \text{either } \omega \text{ does not include an } \mathtt{a} \text{ or } \omega \text{ does not include a } \mathtt{b}\}$$

   that was developed during Lecture #3. This corresponded to the picture shown in Figure 1.

   Modify this DFA — by changing the set F of accepting states — to produce another DFA, M', with the same set of states, start state and transition function — such that the language of M' is

   $$L_3 = \{\omega \in \Sigma^\star \mid \text{either } \omega \text{ includes at least one } \mathtt{a} \text{ or } \omega \text{ include at least one } \mathtt{b} \text{ (or both)}\}.$$

7. Once again, let $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ and let L_3 be the language defined in the previous question. Suppose that we try to design a DFA for L_3 using the ***design process*** described in Lecture #3 — and, in particular, that we start by deciding that an automaton for this language must remember whether ***an*** a ***or a*** b ***(or both) have been seen, so far***.

If you do this then you will end up working with the following subsets of $\Sigma^\star$:

- $S_n = \{\omega \in \Sigma^\star \mid \omega$ only includes c's$\}$
- $S_y = L_3 = \{\omega \in \Sigma^\star \mid \omega$ includes at least one $\mathtt{a}$ or at least one $\mathtt{b}$ (or both)$\}$

(a) Confirm that the ***first sanity check*** is passed. You should review the online notes for Lecture #3, or the summary of the design process that was included as a hand-out, if you do not remember what this is.

(b) Confirm that the ***second sanity check*** is also passed. After this you should be able to identify and work with the following ***states*** that will be included in the automaton being designed.

- q_n: Reached after a string $\omega \in \Sigma^\star$ is processed if and only if $\omega \in S_n$
- q_y: Reached after a string $\omega \in \Sigma^\star$ is processed if and only if $\omega \in S_y$

You should now be able to identify the ***start state*** in the automaton that is being designed, as well.

(c) Confirm that the ***third sanity check*** is also passed. You should now be able to identify the set $F \subseteq Q$ of ***accepting states*** in the automaton being designed.

(d) Confirm that the ***fourth sanity check*** is also passed. This may be surprising, because it was not (initially) true in the example from the lecture.

You should now be able to complete a ***transition table*** that can be used to define the ***transition function*** $\delta : Q \times \Sigma \to Q$ for the automaton being designed.

(e) Finally, draw a picture of the automaton, for the language L_3, whose design has been completed.

This ***will not*** be the same as the automaton (for the same language) from the previous question — this one will only have two states, while the automaton from the previous question had four states, instead.