Homework 3: Regular Equivalences and Nonregularity

CSCI 161

Due 9pm, Thursday, October 8, 2020

Ben Dreyer

Overview. This is your second and final homework focusing on regular languages. The first question recalls the construction for converting an NFA to a DFA; the second has you construct a regular expression equivalent to an FA; the third and fourth have you design FAs equivalent to particular regular expressions, with the fourth requiring you to think beyond the general regex→NFA construction; the last involves proving nonregularity using the pumping lemma (Section 1.4), which will be discussed in class in Monday and Wednesday. *Questions 1-4 are graded on correctness only; full credit for Question 5 depends on logical soundness/clarity.*

Question 0. List any consultations/collaborations for this assignment and elaborate to the extent helpful.

Response. I consulted one of my classmates, Cole, for this assignment.

$N:$

Question 1. For the NFA $N = (Q, \{0, 1\}, \delta, q_0, F)$ depicted above and relabeling $Q = \{1, 2, 3\}$, consider the equivalent DFA $M = (Q', \{0, 1\}, \delta', q_0', F')$ constructed following the general procedure from class, i.e., with $Q' = \mathcal{P}(Q)$, $\delta'(R, a) = E(\bigcup_{r \in R} \delta(r, a))$ for all $R \in Q'$ and $a \in \Sigma$, $q_0' = E(\{q_0\})$, and $F' = \{R \in Q' \mid R \cap F \neq \emptyset\}$, where $E(R)$ denotes the set of states ϵ-reachable from some $r \in R$.

a) How many states are in Q' as defined above?

   8 states, since Q' = P(Q)

b) What is the start state?

   The start state is (1,2) since it is equal to E(1)

c) How many of these states are reachable from the start state?

   Only 2 since this is a DFA, each state only has 2 outputs since the length of our alphabet is 2.

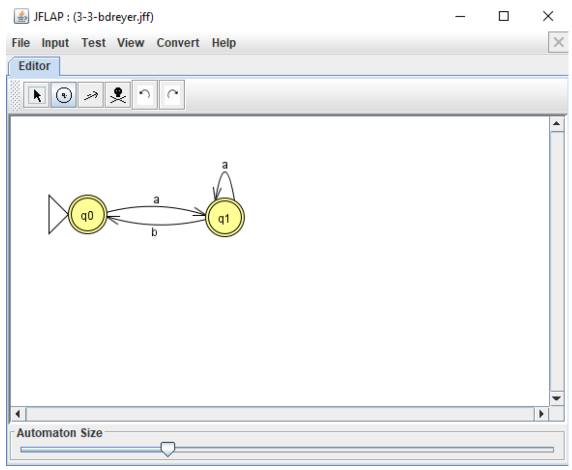
d) What is the label of the state the DFA is in after digesting the characters 10010?

   2

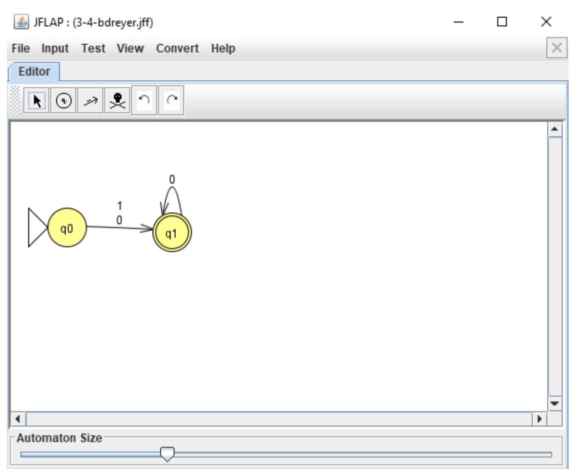
Question 2. Give a regular expression for the language recognized by the machines in Question 1.

Solution. 1*0*1

Question 3. Example 1.56 in the text illustrates the general conversion from a regular expression to an NFA for the regular expression $(ab \cup a)^*$. As with the general conversion from NFAs to DFAs, this general conversion may result in more states than necessary. Submit a JFLAP file containing a two-state NFA that recognizes the language described by $(ab \cup a)^*$, and include a diagram in your pdf.

N : 

Question 4. Submit a JFLAP finite automaton that recognizes the language of $(0 \cup 1) \circ 0^*$, and include a diagram of it in your pdf.

N : 

Question 5. The following language over $\Sigma = \{1, \#\}$ contains #-separated lists of distinct unary values:

$$A = \{x_1 \# x_2 \# \ldots \# x_k \mid k \geq 0, x_i = 1^* \text{ for all } i = 1 \ldots k, x_i \neq x_j \text{ for all } i \neq j\}$$

Use the pumping lemma to show A is not regular. In other words, give a string $s \in A$ and argue that no matter how you partition $s = xyz$ with $|y| > 0, |xy| \leq p$, there exists some $i = 0, 2, 3, \ldots$ such that $xy^i z \notin A$.

Solution. Didn't have enough time to finish this problem