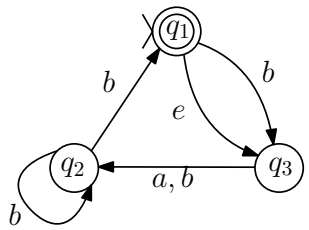
Theory of Computation, Fall 2021
Assignment 2 (Due October 8 Friday 9:35am)

Q1. Run the following NFA on input *abb*. List all the configurations it can reach after consuming all the input symbols. Does it accept *abb*?

(q2,e)
(q3,e)
(q1,e)

yes



Q2. Let A be a regular language over Σ. Let M be an arbitrary DFA that accepts A. Let N be an arbitrary NFA that accepts A.

\Sigma*-L(M)

(a) Let M' be the DFA obtained from M by exchanging the role of final and non-final states. What is $L(M')$?

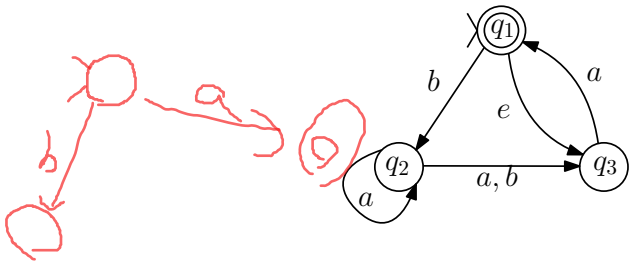
no

(b) Let N' be the NFA obtained from N by exchanging the role of final and non-final states. Is it always true that $L(N) \cap L(N') = \emptyset$? If not, give a counterexample.

Q3. Let $L = \{w \in \{0,1\}^* : w \text{ does not have the substring } 01\}$. Show that L is regular. (Hint: take a look at Q2 and give a finite automata that accepts L)

Q4. Let $L = \{w \in \{a,b,c\} : |w| \geq 1$ and the last symbol of w has appeared at least twice in $w\}$. Construct a NFA to accept L. Your NFA should have no more than 5 states.

Q5. [1, Example 1.41] Convert the following NFA to an equivalent DFA. Give only the portion of the DFA that is reachable from the initial state.



References

[1] Sipser M.. Introduction to the Theory of Computation. CENGAGE Learning (2013)

[2] Lewis H., Papadimitriou C.. Elements of the Theory of Computation. Prentice-Gall (1998)