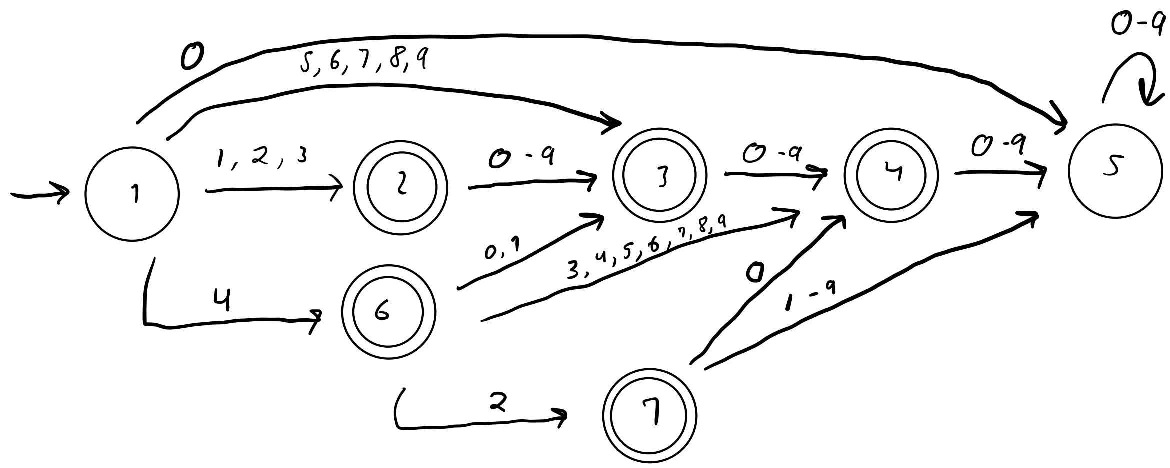
CSCI 2210: Theory of Computation

Problem Set 2 (due 09/23)

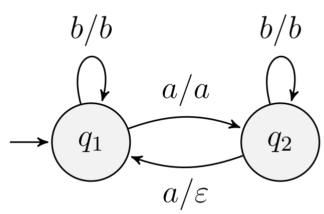
Student: Rafael Almeida *Collaborators:* Cal Thompson

Problem 1. Let $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. Draw a DFA for the language A of strings that is a number between 1 and 420, inclusive, written without leading zeros. Examples, $0 \notin B$, $4 \in B$, $04 \notin B$, $35 \in B$, $045 \notin B$, $454 \notin B$, $410 \in B$.



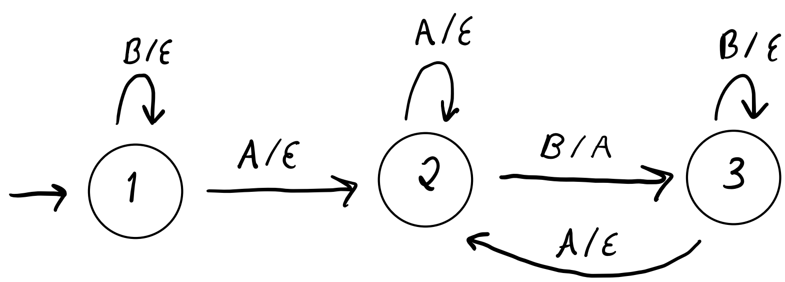
$\diamondsuit$

Problem 2. A deterministic finite-state transducer is a device much like a DFA, except that its purpose is not to accept strings or languages but to transform input strings into output strings. Informally, it starts in a designated initial state and moves from state to state, depending on the input, just as a DFA does. On each step, however, it emits a string of zero or one or more symbols, depending on the current state and the input symbol. The state diagram for a deterministic finite-state transducer looks like that for a DFA, except that the label on an arrow looks like a/w, which means "if the input symbol is a, follow this arrow and output w". For example, the deterministic finite-state transducer over alphabet $\{a, b\}$ shown below outputs all b's in the input string but omits every other a.



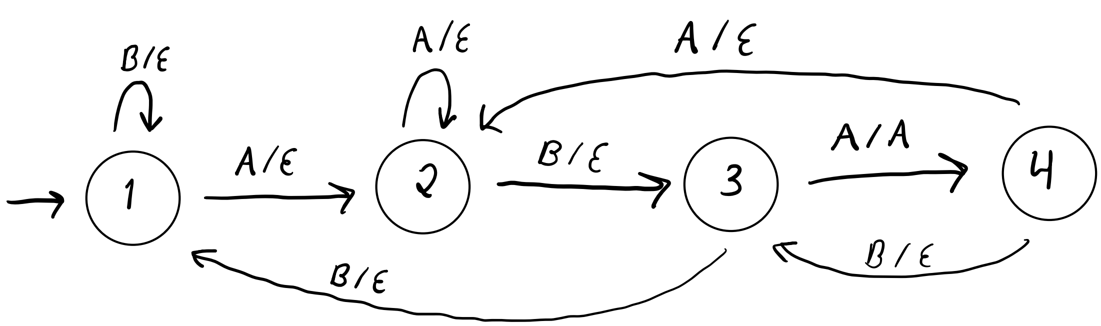
Draw state diagrams for deterministic finite-state transducers over alphabet $\{a, b\}$ that do the following.

(a) On input w, outputs an a for each occurrence of the substring ab in w.



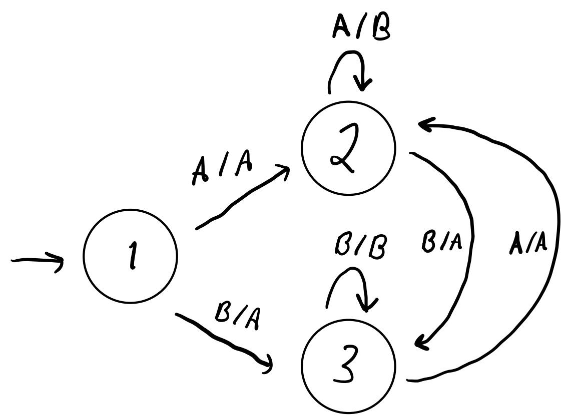
$\diamondsuit$

(b) On input w, outputs an a for each occurrence of the substring aba in w.



$\diamondsuit$

(c) On input w, produce a string of length $|w|$ whose ith symbol is an a if $i = 1$, or if $i > 1$ and the ith and $(i-1)$st symbols of w are different; otherwise, the ith symbol of the output is a b. For example, on input *aabba* the transducer should output *ababa*, and on input *aaaab* it should output *abbba*.



◊

Problem 3. Recall the operation set difference

$$A - B = \{w \mid w \in A, w \notin B\}$$

Show that the class of regular languages is closed under set difference.

$M_A = (Q_A, \Sigma, \delta_A, q_{0_A}, F_A)$, $L(M_A) = L_A$

$M_B = (Q_B, \Sigma, \delta_B, q_{0_B}, F_B)$, $L(M_B) = L_B$

$M_C = (Q_A \times Q_B,$, $L(M_C) = L_A - L_B$

$\Sigma,$

$\delta((p_A, p_B), a) = (\delta_A(p_A, a), \delta_B(p_B, a))$

$(q_{0_A}, q_{0_B}),$

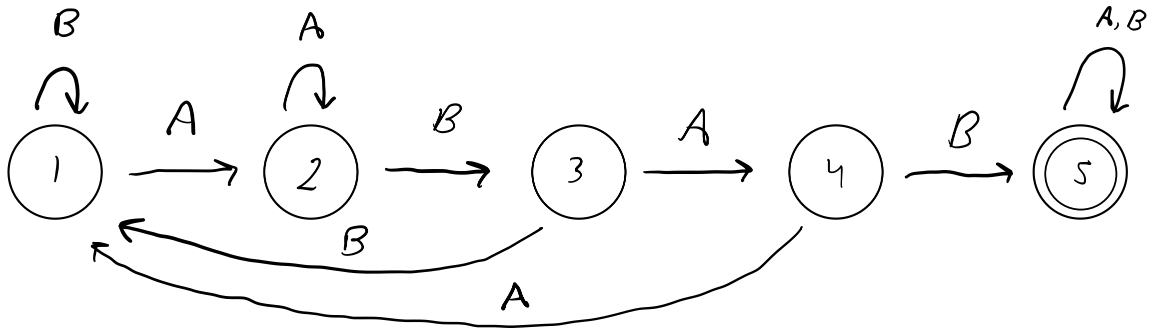
$F_A \times Q_B - Q_A \times F_B)$

◊

The intuition here is that Mc must accept all in La and reject all in Lb (accept all strictly in Ma). Therefore, to build Mc from Ma and Mb, follow the same logic for building a DFA that would accept the intersection of La and Lb (run Ma and Mb in parallel), but make the accepting states of Mc be the states containing only accepting states from Ma. This way, the accepting states of Mc are the states where only strings that are strictly accepted by Ma and not Mb would finish.

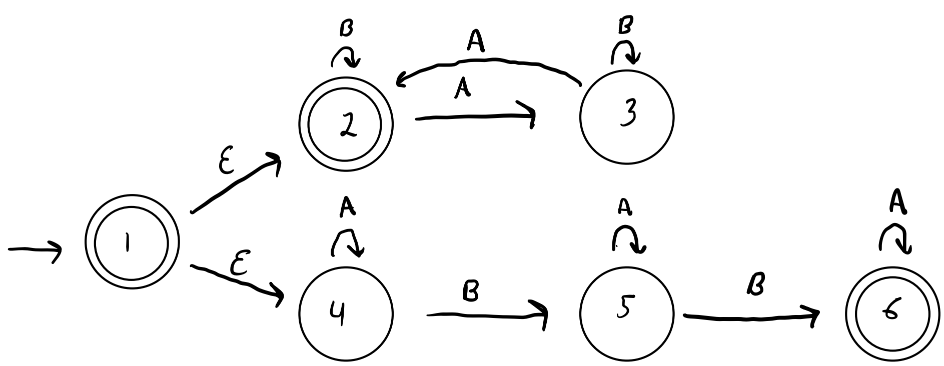
Problem 4. Design NFAs recognizing each of the following languages. Consider $\Sigma = \{a, b\}$.

(a) $L_a = \{w \mid w \text{ contains the substring } abab\}$



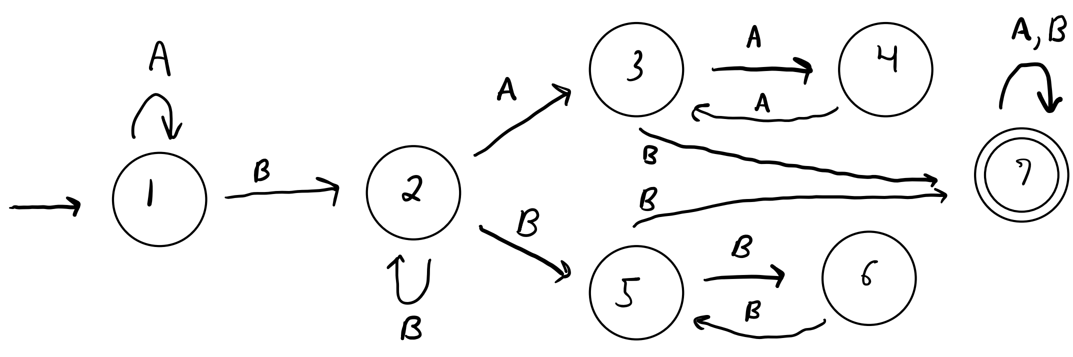
$\diamondsuit$

(b) $L_b = \{w \mid w \text{ contains an even number of } a\text{'s or contains exactly two } b\text{'s}\}$



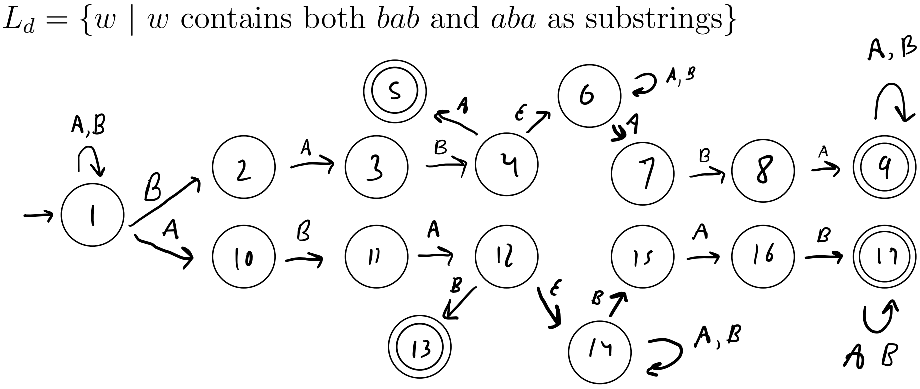
$\diamondsuit$

(c) $L_c = \{w \mid w \text{ contains a pair of } b\text{'s separated by an odd number of } a\text{'s or } b\text{'s}\}$



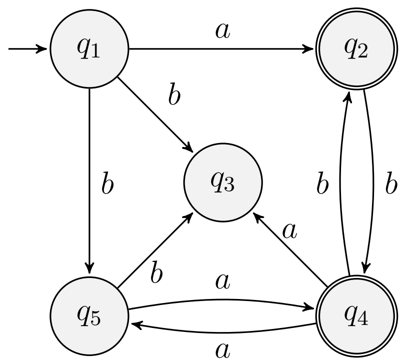
$\diamondsuit$

(d) $L_d = \{w \mid w \text{ contains both } bab \text{ and } aba \text{ as substrings}\}$



◇

Problem 5. Considering the following NFA N, answer the questions below:



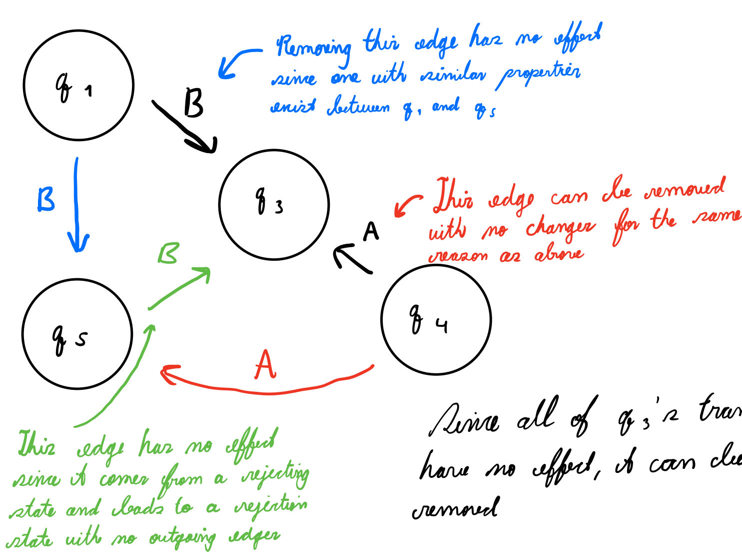
(a) Give 3 examples of strings accepted by N, and 3 examples of strings not accepted. Justify your answer.

Accepted	Rejected
A, q_2	B, $\{q_3, q_5\}$
AB, q_4	BB, q_3
ABB, q_2	BBB, $\emptyset$

For the string to be accepted, it must contain an A.
The set of final states for each string is enumerated.

◇

(b) It is possible to get an NFA equivalent to N by deleting one of the states. Say which state should be deleted and explain why the new NFA is equivalent to N.



Since all of q_3's transitions have no effect, it can be removed

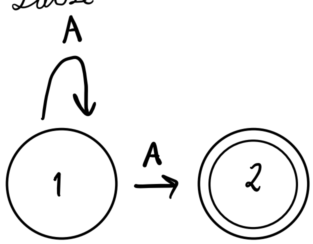
q3 because it contains transitions that are redundant or duplicated. Given that q3 has no outgoing edges, any string that arrives at it is not accepted. Since to arrive at q3 a string must have been in q1, q4, or q5, consider the below justifications for why any string that is not accepted by N and reaches q1, q4, or q5 would still be rejected even if q3 is removed.

◇

Problem 6. Determine if the following statements are true or false. If it is true, give a brief explanation. If it is false, present a counterexample.

(a) For any NFA N_1, swapping the accept and nonaccept states will create an NFA N_2 such that $L(N_2) = \overline{L(N_1)}$, i.e., the language recognized by N_2 is the complement of the language recognized by N_1.

False



For the NFA, M, below, “A” is in L(M). Therefore, “A” must not be in the complement of L(M). Yet, notice that swapping state 2 to not accepting and state 1 to accepting would also accept the string “A”. Therefore, simply toggling the accepting and rejecting states of any NFA is not sufficient to create a new NFA that accepts the complement of the language of the original NFA. ♢

(b) The class of languages recognized by NFAs is closed under complement.

True. Since DFA ≡ NFA and regular languages are closed under complement, the set of languages accepted by NFAs must also be closed under complement. ♢