Harvard University Extension School
Computer Science E-121

Problem Set 1

Due Friday, September 23, 2016 at 11:59pm

Problem set by Walter Thornton

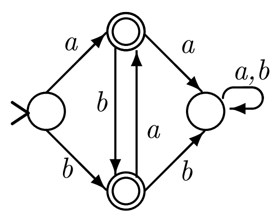
Collaboration Statement: I worked alone and only with course materials

When not stated, assume $\Sigma = \{a, b\}$.

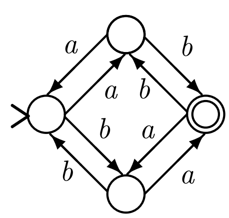
PROBLEM 1 (2+2 points, suggested length of 4 lines)

For each DFA below, informally describe the language it accepts.

(A)



(B)



Solution.

(A) The DFA accepts a single a, a single b, or any combination where a and b alternate. For instance, the DFA accepts a, b, ab, bab, abababa...
(B) The DFa accepts any string of a's and b's where the string ends in some odd number of a's or b's, directly following an odd number of it's opposite. For instance, it accepts, ab, ba, abbb, bbba, aaaabbbaaa

PROBLEM 2 (5+5+(2) points, suggested length of 2/3 page + (2/3 page))

In this problem, we consider the set of nonempty binary strings over the alphabet $\Sigma = \{0, 1\}$, where the highest-order bits are on the left. (For example, the number 8 would be represented in this language by the binary string 1000.)

(A) Draw a DFA that accepts the language of all binary strings which leave a remainder of 2 when divided by 4. *There are lots of ways to draw DFAs in LaTeX. See Piazza for a discussion of some of the ways.*

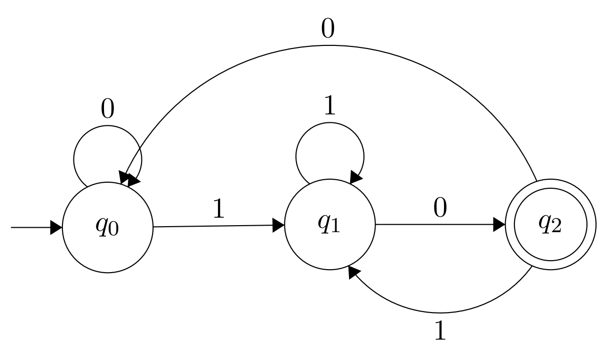
(B) Write out the formal 5-tuple description for the DFA you drew in part (A).

(C) (Challenge!! Not required; worth up to 2 extra credit points.) What is the minimum number of states that a DFA must have to accept exactly the base-k numerals (i.e. non-empty strings over the alphabet $\Sigma_k = \{0, 1, ..., (k-1)\}$) that represent numbers which leave a remainder of m when divided by n. Assume $m < n$, and k and n are co-prime. Please justify your answer for full points.

Note: On every problem set we will provide a challenge problem, generally significantly more difficult than the other problems in the set, but worth only a few points. It is recommended that if you attempt these problems, you do so only after completing the rest of the assignment.

Solution.

(A)



(B) M=$(\{q_0, q_1, q_2\}, \{a, b\}, \delta, q_0, \{q_2\})$ *where* δ *is*

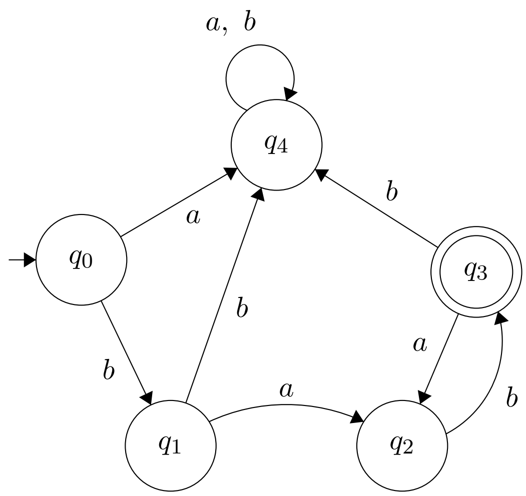
	0	1
q_0	q_0	q_1
q_1	q_2	q_1
q_2	q_0	q_1

(C)

PROBLEM 3 (5 points, suggested length of 1/3 page)

Draw a DFA that only accepts strings that start and end with b, but do not have substring bb. (Try to use as few states as possible.)

Solution.



PROBLEM 4 (1+1+1+1+1+1+1 points, suggested length of 7 lines)

For each of the strings below, determine if it is in the language. (No need to explain.)

$L(a(ab)^*bbb^*)$

(A) abb
(B) $aabb$
(C) $ababab b$

$L(((babb^*a) \cup (aaa\emptyset b) \cup (ab\emptyset^*b^*a))^*)$

(D) aba
(E) $abbbab$
(F) $babaabbababbbaaba$
(G) ε

Solution.

(A) Yes
(B) No
(C) No
(D) Yes
(E) No
(F) Yes
(G) Yes

PROBLEM 5 (4+4 points, suggested length of 1/4 page)

Are the following statements true of false? Justify your answers with a proof or counterexample.

(A) For any languages L_1 and L_2, $(L_1L_2)^*{=}L_1{}^*L_2{}^*$.

(B) If L is a regular language, then the subset of L containing all strings in L of odd length is necessarily regular. Remember that $\Sigma = \{a, b\}$.

Solution.

(A) False
Counterexample:
If $\{\text{aa}\} \in L_1\, and\, \{bb\} \in L_2\ then$

$\{aabbaabb\} \in (L_1L_2)^*$
$\{aabbaabb\} \notin L_1{}^*L_2{}^*$

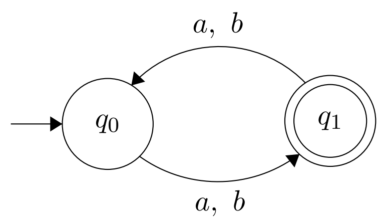
and

$\{aaaabbbb\} \in L_1{}^*L_2{}^*$
$\{aaaabbbb\} \notin (L_1L_2)^*$

$\therefore (L_1L_2)^* \neq L_1{}^*L_2{}^*$

(B) Proof by Construction:

For every NFA, there exists a DFA such that L(DFA)=L(NFA) We can construct a DFA to recognize all strings of L, of odd length, where $\Sigma = \{a, b\}$.
A language is regular if and only if some nondeterministic finite automaton recognizes it.
$\therefore$ the subset of L containing all strings in L of odd length is necessarily regular



PROBLEM 6 (4 points, suggested length of 1/4 page)

Prove using induction that any finite union of regular languages is regular.

Solution.
A language is regular if and only if some nondeterministic finite automaton recognizes it. Therefor if $N_1 = \{Q_1, \Sigma, \delta, q_1, F_1\}$ recognizes $\text{A}_1\, and\, \text{N}_2 = \{Q_2, \Sigma, \delta, q_2, F_2\}$ recognizes A_2 then both N_1 and N_2 are regular languages
Therefore if there is an NFA N that recognizes $A_1 \cup A_2$, than that resulting language A is also regular.
Construct N from N_1 and $N - 2$ where:
$Q = \{q_0\} \cup Q_1 \cup Q_2$
$\Sigma = \Sigma$ of N_1 and N_2

$\delta = \delta_1(q,a)$ remains the same when $q \in Q_1$,
$\delta_2(q,a)$ remains the same when $q \in Q_2$,
$\{q_1, q_2\}$ where $q = q_0$ and $a = \epsilon$,
and $\emptyset$ where $q = q_0$ and $a \neq \epsilon$
q_0,
$F = F_1 \cup F_2$
(from Sipser pg 59)
The preceding NFA N accepts the language A, therefore it is a regular language.

PROBLEM 7 (3+3 points, suggested length of 1/3 page)

Are the following propositions true or false? Justify your answers with a proof or a well-explained counterexample.

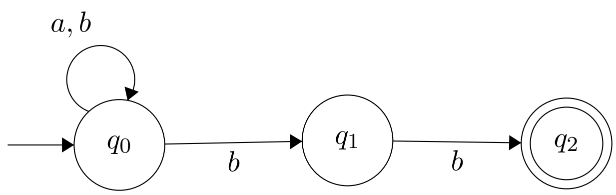
(A) Proposition: Given an NFA $N = (Q, \Sigma, \delta, q_0, F)$, the NFA $N' = (Q, \Sigma, \delta, q_0, Q - F)$ accepts the language $\overline{L(N)} = \Sigma^* - L(N)$.

(B) Setup: Let M be an NFA. We say that M contains a *cycle* if there is a state q and a string x such that if M is in state q and reads string x, M can return to state q. Proposition: If M accepts an infinite language, then M has a cycle.

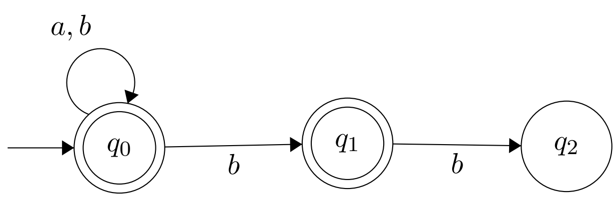
Solution.

(A) False, by counterexample

Construct an NFA that accepts the language $L(N) = \Sigma^* bb$ over the alphabet $\Sigma = \{a, b\}$
The language of N is necessarily finite, as it only accepts upon a string ending with bb.



We can construct another NFA N' where the final states are every state but the original final state from N, or $F_1 = Q - F$



The proposition states that NFA N' would accept the complement of L(N), that is
$\overline{L(N)} = \Sigma^* - L(N)$
This would include infinite strings, as the set of all strings that do not end in bb, is necessarily infinite.

Therefore N' does not accept this language $\overline{L(N)}$, and the proposition is false.

(B) This is true.
According to the definition of a cycle, state q must be returned to at least once.
Assume an NFA with n states and no cycles. Assuming an infinite language, that would mean that in some cases the string would be longer than n.
If the number of states accessed by a particular string is greater than the number of states that exist, then there must be some state that is repeated
Since $n < \infty$, the NFA must cycle to handle strings whose length is greater than the number of states.

Therefore M must have cycles.

PROBLEM 8 (0 points)

Please estimate the number of hours spent on this assignment to the nearest half hour.
Solution.

10, much of which was learning latex