Homework One Solution– CSE 355

Due: 31 January 2011

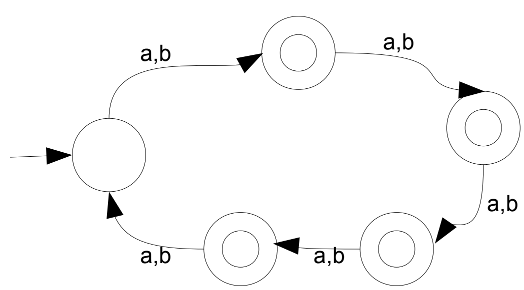
Please note that there is more than one way to answer most of these questions. The following only represents a sample solution.

Problem 1: Linz 2.1.7(b)(c)(g), 2.2.7. and 2.2.11

2.1.7: Find dfa's for the following languages on $\Sigma = \{a, b\}$

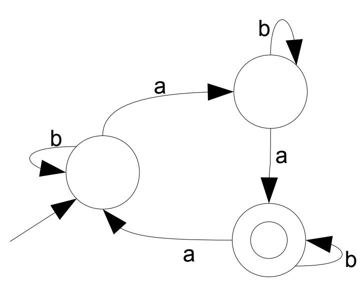
(b): $L = \{w : |w| \, mod \, 5 \neq 0\}$

A dfa for L is given by the following transition graph:



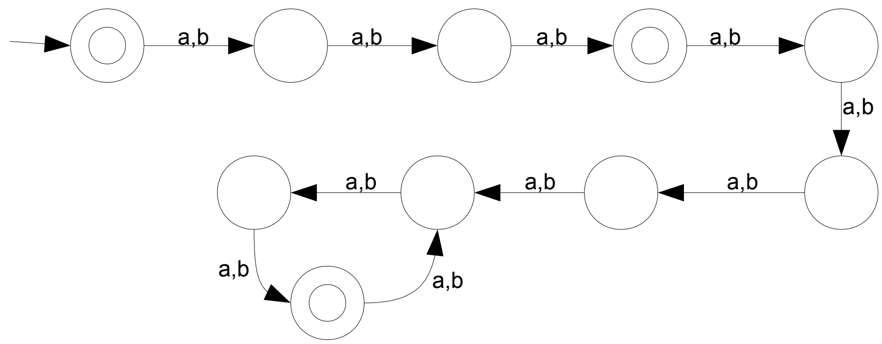
(c): $L = \{w : n_a(w) \, mod \, 3 > 1\}$

A dfa for L is given by the following transition graph:



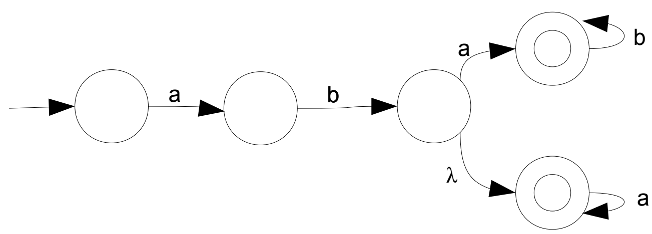
(g): $L = \{w : |w| \; mod \; 3 = 0, \; |w| \neq 6\}$

A dfa for L is given by the following transition graph:



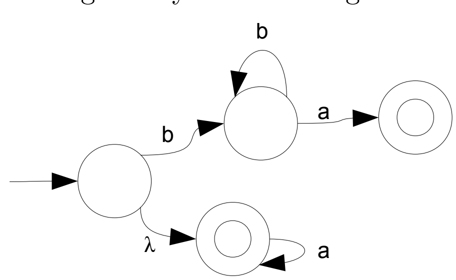
2.2.7: Design an nfa with no more than five states for the set $\{abab^n : n \geq 0\} \cup \{aba^n : n \geq 0\}$.

An nfa for the set is given by the following transition graph:



2.2.11: Find an nfa with foour states for $L = \{a^n : n \geq 0\} \cup \{b^n a : n \geq 1\}$.

An nfa for L is given by the following transition graph:



Problem 2: Linz 2.39 and 2.3.12

2.39: Let L be a regular language that does not contain λ. Show that there exists an nfa without λ-transitions and with a single final state that accept L.

Since L is regular there exists a dfa, $D = (Q, \Sigma, \delta, q_0, F)$, with an associated transition graph, G_D, such that $L(D) = L$. We will construct an nfa $N = (Q \cup \{q_f\}, \Sigma, \delta', q_0, \{q_f\})$ where $q_f \notin Q$ by giving its transition graph G_N as follows:

1. From G_D, remove the final label from every final state (making them nonfinal states).
2. Add a new state q_f and label it as a final state.
3. For every state q_i, if there is a transition from q_i to a state in F on input $a \in \Sigma$, then add a transition from q_i to q_f on input a.

Clearly, N has a single accept state, q_f, and no λ-transitions (since D is a dfa and we did not add any λ-transitions in our construction of N). We will now show that $L(N) = L$. First note that since $\lambda \notin L$, every $w \in L$ can be written as $w = va$ for some $v \in \Sigma^*$ and an $a \in \Sigma$.

Now, $w = va \in L$ iff there is a walk on G_D labeled with w from q_0 to q_i with $q_i \in F$
iff there is a walk on G_D labeled with v from q_0 to q_j and a transition from q_j to q_i on input a
iff there is a walk on G_N labeled with v from q_0 to q_j and a transition from q_j to q_f on input a
(since every transition in G_D is a transition in G_N and from step (3) in the construction of G_N)
iff there is a walk on G_N labeled with w from q_0 to q_f
iff $w \in L(N)$.

Thus, $w \in L$ iff $w \in L(N)$. Therefore we conclude that $L(N) = L$ and that for any regular language that does not contain λ, there exists an nfa without λ-transitions and with a single final state that accept L.

2.3.12: Show that if L is regular, so is L^R.

Since L is a regular language, we can construct a corresponding dfa, N, such that $L(N) = L$ (For every regular language, there is a corresponding dfa, by definition, and for every dfa, there is an equivalent nfa).

By definition, L^R consists of all strings in language L in reverse order. We will construct a nfa, N_R, representing L^R such that $L(N_R) = L^R$. N_R will contain an additional start state with λ-transitions to the final states of N. The direction of every transition in N is reversed. Also, the start state of N will be the final state of N_R. The construction of nfa N_R is as follows:

Let $N = (Q, \Sigma, \delta, q_n, F)$
$N_R = (Q \cup \{q_0\}, \Sigma, \delta_r, q_r, \{q_n\})$
Set of states of N_R = set of states of N along with $q_0 = Q \cup \{q_r\}$
Σ = alphabet of N_R = same as N
q_r = start state of N_R
$\{q_n\}$ = set of final states of N_R = start state of N
Transition function:
$\delta_r(q, a) = \{q_1 : \delta(q_1, a) = q\}$
$\delta_r(q_r, \lambda) = F$

$\delta_r(q_r, a) = \emptyset$, if $a \neq \lambda$

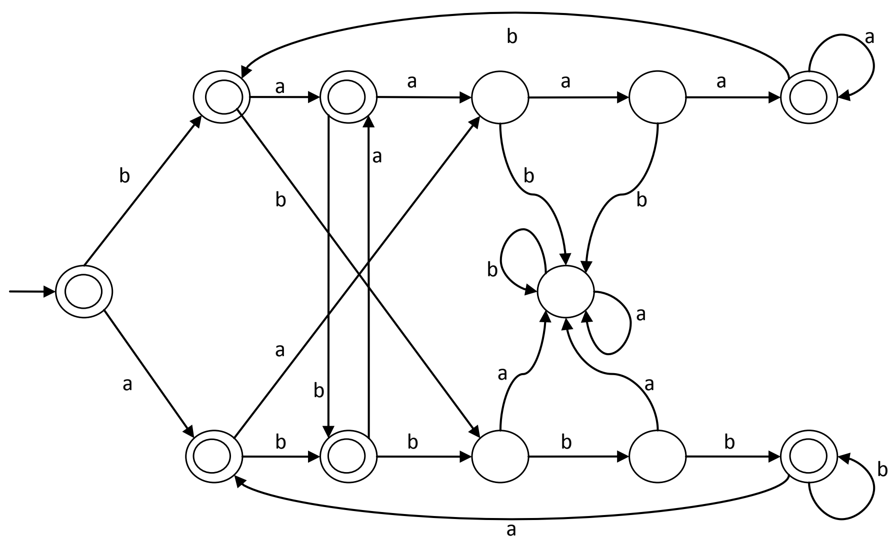
Now we will show that $L^R = L(N_R)$. $w \in L^R$ iff $w^R \in L$ iff there is a walk on the transition graph of N with label w^R from q_n to some $q_i \in F$ iff there is a walk on the transition graph of N_R from q_r to q_i with label λ and a walk from q_i to q_n with label w (Following the reverse of every transition in the original graph) iff $w \in L(N_R)$.

Since L_R can be represented by a nfa, it is regular (by equivalence of nfa to dfa, and dfa to regular language).

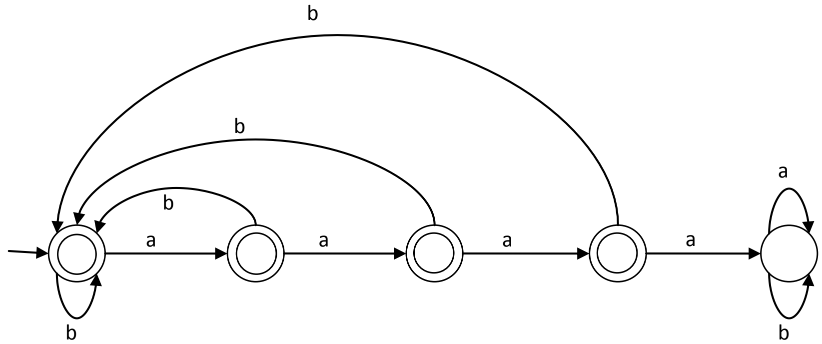
Problem 3: Linz 2.1.8

2.1.8: A run in a string is a substring of length at least two, as long as possible and consisting entirely of the same symbol. For instance, the string $abbbaab$ contains a run of b's of length three and a run of a's of length two. Find dfa's for the following languages on $\{a, b\}$.

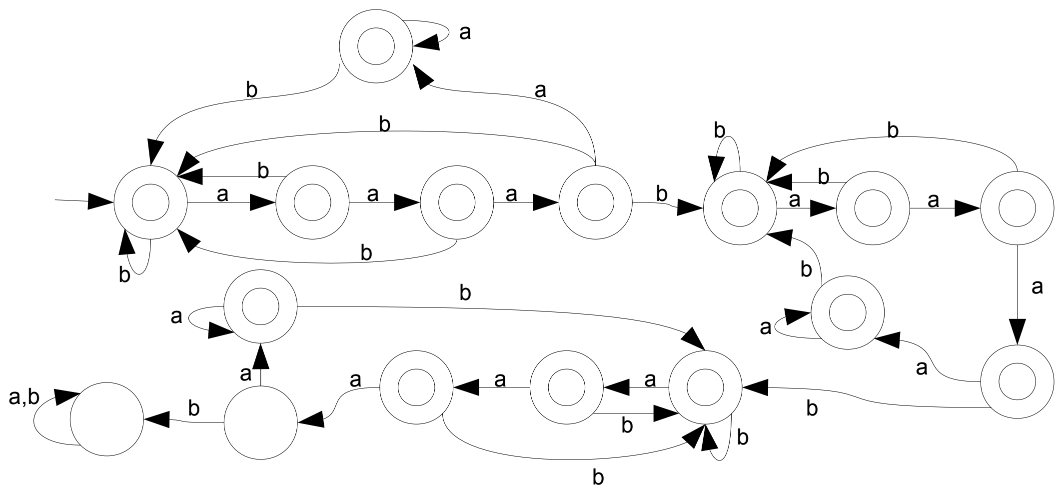
(a): $L = \{w : w$ **contains no runs of length less than four**$\}$.



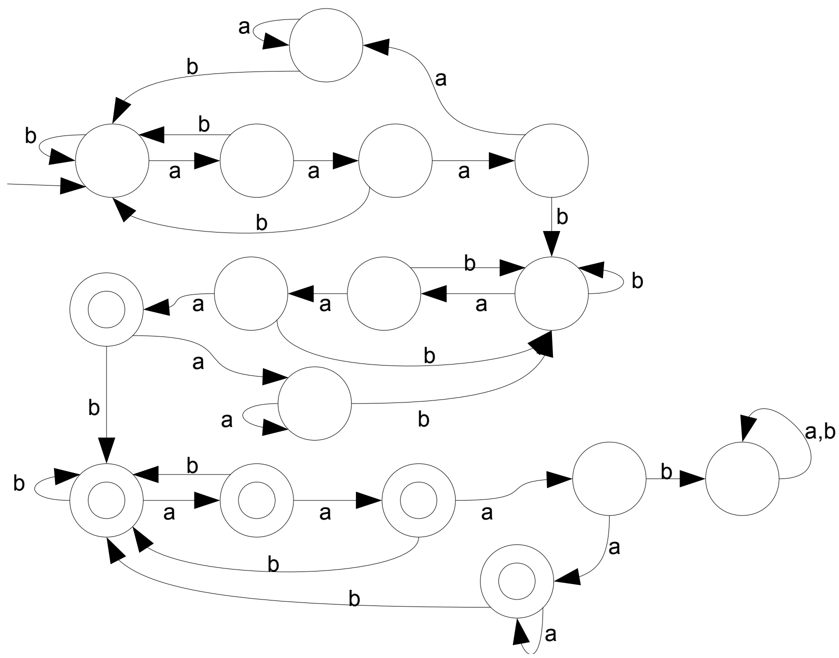
(b): $L = \{w :$ **every run of** a'**s has length either two or three**$\}$.



(c): $L = \{w :$ **there are at most two runs of** a'**s of length three**$\}$.



(d): $L = \{w :$ **there are exactly two runs of** a'**s of length 3**$\}$.



Problem 4: Linz 2.2.22

2.2.22: Let L **be a regular language on some alphabet** Σ**, and let** $\Sigma_1 \subset \Sigma$ **be a smaller alphabet. Consider** L_1**, the subset of** L **whose elements are made up only of symbols from** Σ_1**, that is,**

$$L_1 = L \cap \Sigma_1^*.$$

Show that L_1 **is also regular.**

Since L is a regular language, there should be a dfa, N, representing L such that $L(N) = L$, where $N = (Q, \Sigma, \delta, q_0, F)$.

Since L_1 is made up of strings with alphabets from Σ_1, $\Sigma_1 \subset \Sigma$, and L_1 is a subset of L, L_1 contains only strings that are accepted by L as well. We can construct a dfa, M, for L_1 as follows:

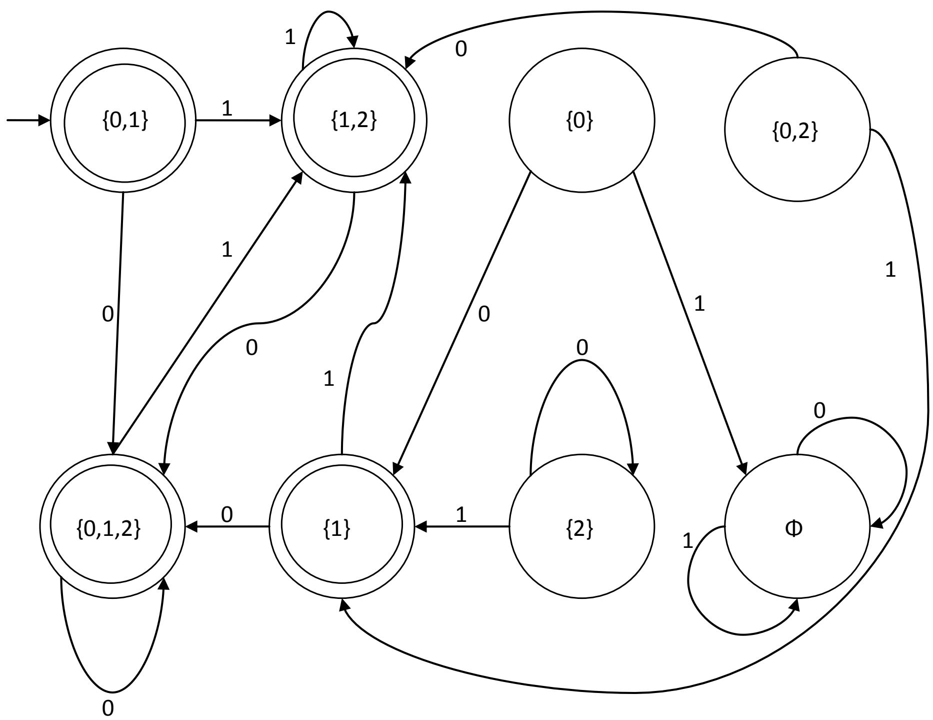
1. From the transition graph of N, remove every transition that is labeled with some $a \notin \Sigma_1$.

Now we will show that $L(M) = L_1$. $w = a_1 a_2 \ldots a_n \in L_1$ iff there is a walk on the transition graph of N with label w from q_0 to some $q_i \in F$ and every $a_i \in \Sigma_1$ iff there is a walk on the transition graph of M from q_0 to q_i with label w (it will be the exact same path as it was in N) iff $w \in L(M)$.

Since L_1 can be represented by a dfa, it is regular.

Problem 5: Linz 2.3.3 and 2.3.8

2.3.3: Convert the following nfa into an equivalent dfa (see textbook for the diagram).



2.3.8: Find an nfa without λ-transitions and with a single final state that accepts $L = \{a\} \cup \{b^n : n \geq 1\}$.

Noting that $\lambda \notin L$, we can use the technique given in 2.3.9 (Problem 2) and we get the nfa given by the following transition graph: