CS F351 Theory of Computation
Tutorial-4

Note: $*$ marked problems can be left to the students to try after the tutorial.

Problem 1 Construct NFA for the following languages:

$L_3 = \{\omega \in \{a_0, a_1, a_2\}^* \mid \omega$ is a string in which at least one a_i occurs even number of times (not necessarily consecutive), where $0 \leq i \leq 2\}$.

Solution:

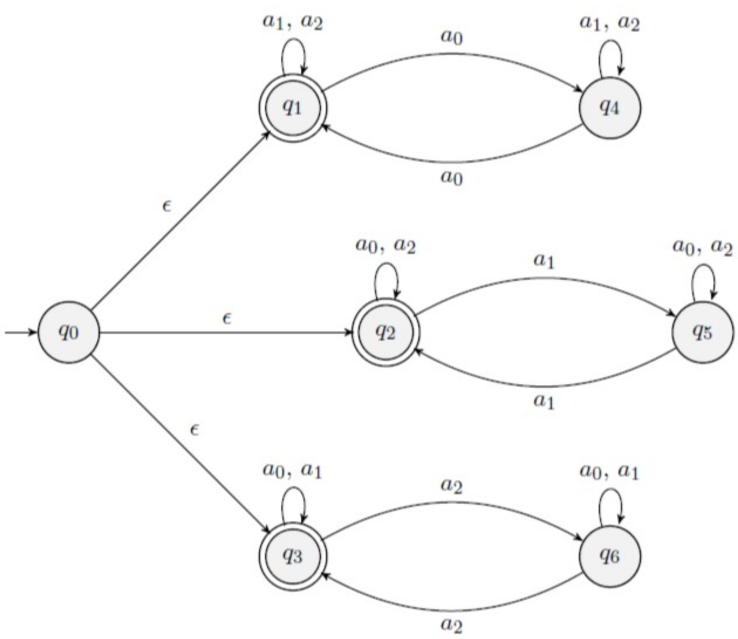


Figure 1: Solution to Problem 2.1.

Problem 2 Let $A \subseteq \Sigma^*$ be a language. Show that if A is regular then A^R is also regular where $A^R = \{x^R \mid x \in A\}$.

Solution Idea: Let $M = (Q = \{q_0, q_1, \ldots, q_k\}, \Sigma, \delta, q_0, F)$ be an NFA such that $L(M) = A$.

To construct an NFA M' for A^R, in M reverse the direction of the edges, make q_0 as the final state. If F has a unique state, then make it as the starting state for M', otherwise create a new start state s and put an ϵ-transition from s to each state in F (now a unique start state is defined for M').

Problem 3 $(*)$ For languages A and B, let the **perfect shuffle** of A and B be a language

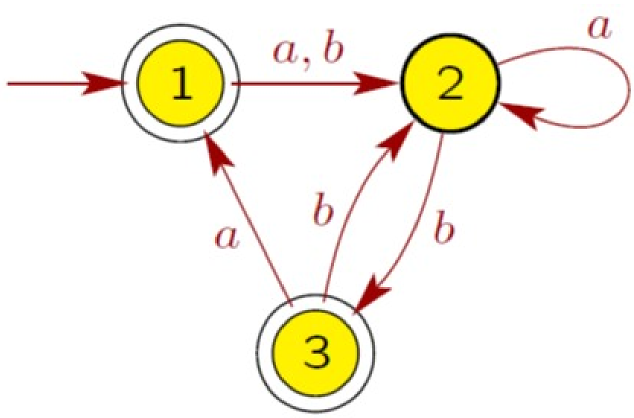
$\{w \mid w = a_1b_1a_2b_2 \ldots a_kb_k,$ where $a_1a_2 \ldots a_k \in A$ and $b_1b_2 \ldots b_k \in B,$ each $a_i, b_i \in \Sigma\}$.

Show that if A and B are regular then perfect shuffle of A and B is also regular.

Solution:

Please see a solution on page 2, problem 5

Problem 4 *Use the procedure discusses in the class to convert the following DFA to regular expression.*



Solution:

Please see solution on Problem 2.

Problem 5 *Use the procedure discusses in the class to convert the regular expression* $(((00)^*(11)) + 01)^*$ *into an NFA (with* ϵ*-transitions).*

Solution:

Please see solution on Problem 1.
