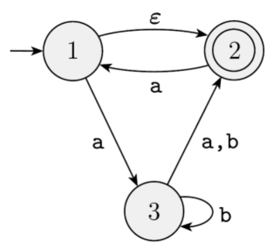
Assignment 3

Computer Science 235

Reading. Section 1.3

1) Use the construction given in Theorem 1.39 in Sipser to convert the following nondeterministic finite automaton to an equivalent deterministic finite automaton.



2) Give regular expressions generating the following languages.

a) {*w* | *w* begins with a 1 and ends with a 0}

b) {*w* | *w* contains at least three 1s}

c) {*w* | *w* contains the substring 0101}

d) {*w* | *w* has length at least 3 and its third symbol is a 0}

e) {*w* | *w* starts with a 0 and has odd length, or starts with a 1 and has even length}

3) Use the procedure described in Lemma 1.55 in Sipser to convert the following regular expressions to nondeterministic finite automata.

a) $(((00)^*(11)) \cup 01)^*$

b) $\emptyset^*$

4) For each of the following languages, give two strings that are members and two strings that are not members - a total of four strings for each part. Assume the alphabet $\Sigma = \{a, b\}$.

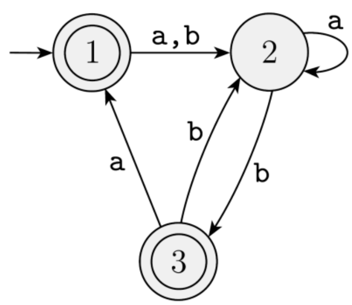
a) $a(ba)^*b$

b) $a^* \cup b^*$

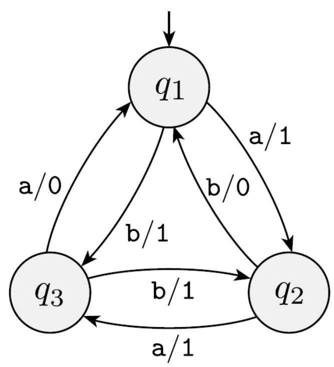
c) $\Sigma^*a\Sigma^*b\Sigma^*a\Sigma^*$

d) $(\varepsilon \cup a)b$

5) Use the procedure described in Lemma 1.60 in Sipser to convert the following finite automaton to a regular expression.



6) A ***finite state transducer*** (FST) is a type of deterministic finite automaton whose output is a string and not just *accept* or *reject*. The following is a state diagram of a finite transducer T.



Each transition of an FST is labeled with two symbols, one designating the input symbol for that transition and the other designating the output symbol. The two symbols are written with a slash, /, separating them. In T, the transition from q_1 to q_2 has input symbol a and output symbol 1. When an FST computes on an input string w, it takes the input symbols $w_1 \cdots w_n$ one by one and, starting at the start state, follows the transitions by matching the input labels with the sequence of symbols $w_1 \cdots w_n = w$. Every time it goes along a transition, it outputs the corresponding output symbol. For example, on input abbb, T enters the sequence of states q_1, q_2, q_1, q_3, q_2 and produces output 1011. Give (a) the sequence of states entered and (b) the output produced by T for the input bbab.

7) Give the formal definition of the FST model (see previous problem for definition of FST), following the pattern in Definition 1.5 in Sipser. Assume that an FST has an input alphabet Σ and an output alphabet Γ but not a set of accept states. Include a formal definition of the computation of an FST, following the pattern on page 40 in Sipser. (Hint: An FST is a 5-tuple. Its transition function is of the form $\delta: Q \times \Sigma \rightarrow Q \times \Gamma$.)

8) Recall that string x is a ***prefix*** of string y if a string z exists where $xz = y$, and that x is a ***proper prefix*** of y if in addition $x \neq y$. Consider the following operation on a language A:

$$NOEXTEND(A) = \{ w \in A \mid w \text{ is not the proper prefix of any string in } A \}$$

Show that the class of regular languages is closed under the *NOEXTEND* operation.

9) For languages A and B, let the ***perfect shuffle*** of A and B be the language

$$\{w \mid w = a_1b_1\cdots a_kb_k, \text{ where } a_1\cdots a_k \in A \text{ and } b_1\cdots b_k \in B, \text{ with each } a_i, b_i \in \Sigma\}.$$

Show that the class of regular languages is closed under perfect shuffle.