Problem Set 2

Fall 11

Due: 27th September, 2011, 11:00 am before class begins

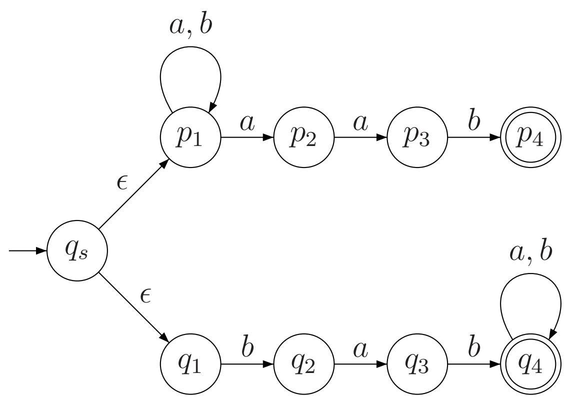
Please follow the homework format guidelines posted on the class web page:

`http://www.cs.illinois.edu/class/fa11/cs373/`

Also, note that Problem 6 is an extra credit question.

1. [**Category**: NFA Comprehension, **Points**: 20]

Consider the following NFA M.



(a) Formally show that M accepts the string $w_1 = abaaab$ and string $w_2 = babaab$.

(b) Give a formal definition of the language that M recognizes. Briefly describe why M recognizes it.

Solution:

(a) Note that $w_1 = abaaab = \epsilon abaab$. A sequence of states accepting this representation is $q_s, p_1, p_1, p_1, p_2, p_3, p_4$. In this sequence, q_s is the initial state and p_4 is a final state. Moreover, $p_1 \in \delta(q_s, \epsilon)$, $p_1 \in \delta(p_1, a)$, $p_1 \in \delta(p_1, b)$, $p_2 \in \delta(p_1, a)$, $p_3 \in \delta(p_2, a)$, and $p_4 \in \delta(p_3, b)$. Therefore, by definition of acceptance in NFA, M accepts w_1.

Note that $w_2 = babaab = \epsilon babaab$. A sequence of states accepting this representation is $q_s, q_1, q_2, q_3, q_4, q_4, q_4, q_4$. In this sequence, q_s is the initial state and q_4 is a final state. Moreover, $q_1 \in \delta(q_s, \epsilon)$, $q_2 \in \delta(q_1, b)$, $q_3 \in \delta(q_2, a)$,

$q_4 \in \delta(q_3, b)$, $q_4 \in \delta(q_4, a)$, $q_4 \in \delta(q_4, a)$, and $q_4 \in \delta(q_4, b)$. Therefore, by definition of acceptance in NFA, M accepts w_2.

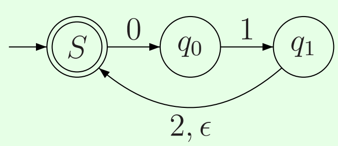
(b) The language of M is the set of strings over $\Sigma = \{a, b\}$ that either begins with bab or ends with aab.

2. [**Category**: NFA Construction, **Points**: 20]

Construct a non-deterministic finite automata that accepts the language $\{01, 012\}^*$ over the alphabet, $\{0, 1, 2\}$. Your automata should contain only three states.

Hint: Think nondeterminism, and ϵ is your friend.

Solution:



3. [**Category**: Construction, **Points**: 20]

For a string w, the reverse of w is defined as the string obtained by reading s from right to left, denoted by w^{-1}. For example, if $w = abc$, then $w^{-1} = cba$; if $w = abab$, then $w^{-1} = baba$.

For a language L, the reverse of L is defined as the language

$$reverse(L) = \{w^{-1} \mid w \in L\}$$

Let $A = \{Q, \Sigma, \delta, q_0, F\}$ be a DFA accepting L, construct an NFA B with no more than $|Q| + 1$ states that will accept $reverse(L)$. Give the formal definition of B (in tuple notation, no diagram). You should also argue how/why this NFA works (intuitive explanation is enough).

Solution:

Intuitively, B can be constructed by reverse each transition in A, and then exchange the initial state and the final states. Note that the resulting automaton is non-deterministic, in general. (If $\delta(p, a) = q$ and $\delta(p', a) = q$, then reverse transitions from q will non-deterministically go to p_1 and p_2.) However, there is a problem. Since A might has multiple final states, the reversed automaton will have multiple initial states, which is not allowed. Thanks to the nondeterminism in an NFA, we can *guess* the initial state. The idea is as follows: We introduce a new state q_s, which is the unique initial state. Also we add an ϵ-transition from q_s to each state that is final in A).

Formally, if $A = \{Q, \Sigma, \delta, q_0, F\}$, we construct $B = \{Q \cup \{q_s\}, \Sigma, \delta_{rev}, q_s, \{q_0\}\}$, where q_s is a new state that is not in Q, and where δ_{rev} is defined as

- $\delta_{rev}(q_s, \epsilon) = F$
- $\delta_{rev}(q_s, a) = \emptyset$ for every $a \in \Sigma$
- $\delta_{rev}(q, \epsilon) = \emptyset$ for every $q \in Q$
- $\delta_{rev}(q, a) = \{q' \mid \delta(q', a) = q\}$ for every $q \in Q$

Then B accepts *reverse*(L).

4. [**Category**: Regular Expressions, **Points**: 4+4+4+8]

Give a regular expression for each of the following languages; the alphabet is $\{a, b\}$.

- The set of all words that end with a b.
- The set of all words that begin with aa and end with ab.
- The set of all words such that every occurrence of a is immediately followed by a b.
- The set of all words such that the number of changes from a to b is the same as the number of changes from b to a when read left to right.
(E.g., $aabbbabbbba$ is in the language, as there are two places where a's change to b's and two places where b's change to a's; however, $aabbbab$ is not in the language as a's change to b's twice, while b's change to a's only once).

Solution:

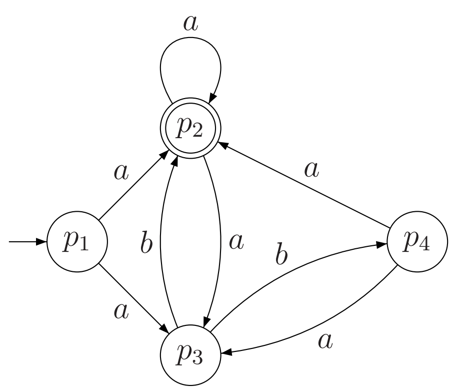
- $(a + b)^*b$
- $aa(a + b)^*ab$
- $(b + ab)^*$

- $(a.a^*.b.b^*)^*.(a.a^*) + (b.b^*.a.a^*)^*.(b.b^*) + \epsilon$

  The first expression describes those words in the language that start with an a, while the second expression describes those words that start with a b. Epsilon is also in the language.

5. [**Category**: NFA to DFA Conversion, **Points**: 20]

   Convert the following NFA to a DFA using the subset construction, and show the state diagram.

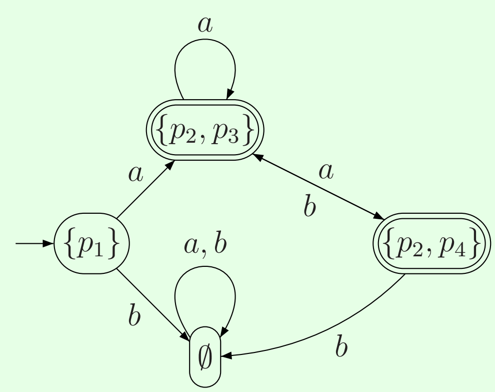


You can check your answer (if you wish) by feeding a DFA to the website:

`http://pub.ist.ac.at/automata_tutor/solve?pid=16`

However, the site does not check if you are describing a DFA; also note that you will lose points if you do not follow the subset construction.

Solution:



6. [**Category**: Extra Credit:, **Points**: 20]

Give a language L over the alphabet $\Sigma = \{a, b\}$ such that any DFA accepting L requires at least 3 final states. Prove that the language L you give has this property.