CMPS 130
Computational Models
Spring 2016
Midterm Exam 1 Solutions

1. (20 Points) Let S be a set. Prove that no function $f : S \to 2^S$ is onto. (Recall that 2^S denotes the power set of S, i.e. the set of all subsets of S.)
   Proof:
   Assume $f : S \to 2^S$ is onto. Define $A \subseteq S$ by $A = \{ x \in S \mid x \notin f(x) \}$. Since f is supposedly onto, there exists $y \in S$ such that $f(y) = A$. Either $y \in A$ or $y \in S - A$. If $y \in A$ then $y \in f(y)$, so by the definition of A we have $y \notin A$, a contradiction. On the other hand if $y \notin A$ then $y \notin f(y)$ so again by definition of A we have $y \in A$, another contradiction. These contradictions show that our assumption was false, and therefore no such function f can exist. ///

2. (20 Points) Let Σ be a finite alphabet and let $M = (Q, \Sigma, q_0, A, \delta)$ be a finite state automaton. Write down the following recursive definitions.

   a. (5 Points) Σ^* the set of all strings over Σ.
   Definition:
   (1) $\lambda \in \Sigma^*$
   (2) If $x \in \Sigma^*$ and $\sigma \in \Sigma$, then $x\sigma \in \Sigma^*$.

   b. (5 Points) $r : \Sigma^* \to \Sigma^*$ the reversal function.
   Definition:
   (1) $r(\lambda) = \lambda$
   (2) If $x \in \Sigma^*$ and $\sigma \in \Sigma$, then $r(x\sigma) = \sigma\, r(x)$

   c. (5 Points) $\delta^* : Q \times \Sigma^* \to Q$ the extended transition function.
   Definition:
   (1) If $q \in Q$, then $\delta^*(q, \lambda) = q$
   (2) If $x \in \Sigma^*$, $\sigma \in \Sigma$ and $q \in Q$, then $\delta^*(q, x\sigma) = \delta(\delta^*(q, x), \sigma)$

   d. (5 Points) $\mathscr{F} \subseteq 2^{\Sigma^*}$ the set of all finite languages over Σ.
   Definition:
   (1) $\varnothing$, $\{\lambda\}$, $\{\sigma\} \in \mathscr{F}$ (for any $\sigma \in \Sigma$)
   (2) If $L_1, L_2 \in \mathscr{F}$, then $L_1 \cup L_2 \in \mathscr{F}$
   (3) If $L_1, L_2 \in \mathscr{F}$, then $L_1 L_2 \in \mathscr{F}$

3. (20 Points) Let Σ be a finite alphabet, $M = (Q, \Sigma, q_0, A, \delta)$ be a finite state automaton, and let $p \in Q$. Prove that for all strings $x, y \in \Sigma^*$: $\delta^*(p, xy) = \delta^*(\delta^*(p, x), y)$. Hint: write this as

$$\forall y \ \boxed{\forall x \ \ \delta^*(p, xy) = \delta^*(\delta^*(p, x), y)}$$

and proceed by structural induction on *y*. To do this let $P(y)$ be the boxed statement and prove: I. $P(\lambda)$ is true, and II. if $P(y)$ holds, then so does $P(y\sigma)$ for any $\sigma \in \Sigma$.

Proof:

I. Show $P(\lambda)$:

$$\begin{aligned} \delta^*(p, x\lambda) &= \delta^*(p, x) && \text{(since } x = x\lambda\text{)} \\ &= \delta^*(\delta^*(p, x), \lambda) && \text{(by (1) in the recursive definition of } \delta^*\text{)} \end{aligned}$$

II. Show $P(y) \Rightarrow P(y\sigma)$:

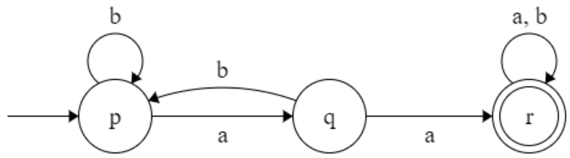
Assume $P(y)$ holds for some $y \in \Sigma^*$, i.e. assume: $\forall x \ \ \delta^*(p, xy) = \delta^*(\delta^*(p, x), y)$. We must show that $P(y\sigma)$ holds for any $\sigma \in \Sigma$, i.e. show: $\forall x \ \ \delta^*(p, x(y\sigma)) = \delta^*(\delta^*(p, x), y\sigma)$. We have

$$\begin{aligned} \delta^*(p, x(y\sigma)) &= \delta^*(p, (xy)\sigma) && \text{(by the associative law)} \\ &= \delta(\delta^*(p, xy), \sigma) && \text{(by (2) in the recursive definition of } \delta^*\text{)} \\ &= \delta(\delta^*(\delta^*(p, x), y), \sigma) && \text{(by the induction hypothesis)} \\ &= \delta^*(\delta^*(p, x), y\sigma) && \text{(by (2) in the recursive definition of } \delta^*\text{)} \end{aligned}$$

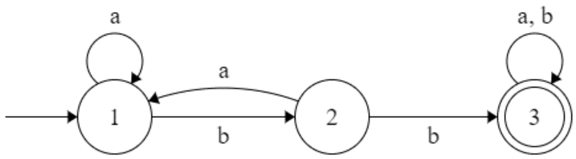
Therefore $P(y)$ holds for all $y \in \Sigma^*$, i.e. $\delta^*(p, xy) = \delta^*(\delta^*(p, x), y)$ for all $x, y \in \Sigma^*$. ///

4. (20 Points) Let L_1 and L_2 be the languages over $\{a, b\}$ accepted by the FAs M_1 and M_2 pictured below, respectively.

M_1:

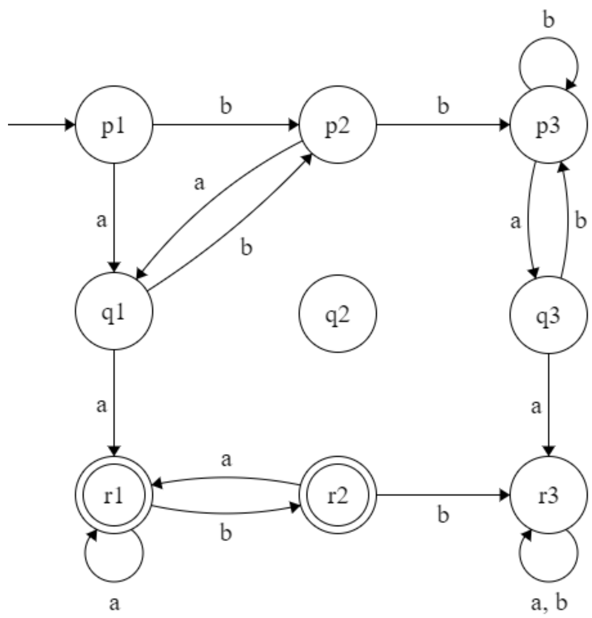


M_2:



Use the product construction to draw an FA accepting the language $L_1 - L_2$. Give simple verbal descriptions of the languages L_1, L_2 and $L_1 - L_2$. (No justification is necessary for either the drawing or the descriptions.)

Solution:



$L_1 =$ { strings containing aa }
$L_2 =$ { strings containing bb }
$L_1 - L_2 =$ { strings containing aa but not containing bb}

5. (20 Points) Let L_1 and L_2 be languages over a finite alphabet Σ.

a. (10 Points) Prove that $(L_1 \cap L_2)^* \subseteq L_1^* \cap L_2^*$.

Proof:

Let $x \in (L_1 \cap L_2)^*$. Then $x \in (L_1 \cap L_2)^n$ for some $n \geq 0$. Thus $x = x_1 x_2 \ldots x_n$ for some $x_i \in L_1 \cap L_2$ ($1 \leq i \leq n$). But then both $x_i \in L_1$ and $x_i \in L_2$ ($1 \leq i \leq n$). Therefore both $x \in L_1^n$ and $x \in L_2^n$, hence $x \in L_1^*$ and $x \in L_2^*$. Therefore $x \in L_1^* \cap L_2^*$, whence $(L_1 \cap L_2)^* \subseteq L_1^* \cap L_2^*$. ///

b. (10 Points) Give an example of specific Σ, L_1 and L_2 for which $(L_1 \cap L_2)^* \neq L_1^* \cap L_2^*$.

Example:

Let $\Sigma = \{a, b\}$, $L_1 = \{a, ba\}$ and $L_2 = \{ab, a\}$. Then $aba \in L_1^2 \subseteq L_1^*$ and $aba \in L_2^2 \subseteq L_2^*$, whence $aba \in L_1^* \cap L_2^*$. But $L_1 \cap L_2 = \{a\}$ so that $(L_1 \cap L_2)^* = \{ a^n \mid n \geq 0 \}$. Thus $aba \notin (L_1 \cap L_2)^*$ showing that $(L_1 \cap L_2)^* \neq L_1^* \cap L_2^*$. ///

Note: there are many other possible examples illustrating $(L_1 \cap L_2)^* \neq L_1^* \cap L_2^*$.