Math 448 Fall 2011, Computer Algebra
Instructor: Sreekar M. Shastry
Solutions to the Mid Semester Examination
12-Oct-2011 1400-1530 in Room C304 of HR4

- ⋆ There are 6 problems. Each problem is worth 5 points. The maximum score is 30 points.
- ⋆ Clearly state the results you invoke.

1. Let $M = (S, A, \mu, s_0, Y)$ be a deterministic finite state automaton over the alphabet A, where S is the set of states, Y is the set of accept states, and s_0 is the start state. Recall the definition of $\widehat{\mu}$ given inductively as follows: $\widehat{\mu}(q, \varepsilon) := q$ and $\widehat{\mu}(q, ua) := \mu(\widehat{\mu}(q, u), a)$ for $u \in A^*, a \in A$.

Suppose that there exists an $a \in A$ such that for all $q \in S$ we have $\mu(q, a) = q$.

(a) Show that $\widehat{\mu}(q, a^n) = q$ for all $n \geqslant 0$ where a^n is the string consisting of n a's.

(b) Show that either $\{a\}^* \subseteq L(A)$ or $\{a\}^* \cap L(A) = \varnothing$.

Solution. (a) Proof by induction. We have been given the base case $\widehat{\mu}(q, a) = \mu(q, a) = q$. Assume the statement is true for $n-1$. Then $\widehat{\mu}(q^n, a) = \mu(\widehat{\mu}(q, a^{n-1}), a) = \mu(q, a) = q$, as required.

(b) If $\{a\}^* \cap L(M) \neq \varnothing$ then $a^k \in L(M)$ for some k and $\mu(s_0, a^k)$ is an accept state. This implies that the start state is an accept state since $\widehat{\mu}(s_0, a) = s_0$ by assumption. Thus for any $j \geqslant 0$ we have $\mu(s_0, a^j) = s_0$ is an accept state and therefore $\{a\}^* \subset L(M)$, as required.

2. We are given a deterministic automaton $M = (S, A, \mu, s_0, Y)$.

(a) Show that

$$\widehat{\mu}(q, xy) = \widehat{\mu}(\widehat{\mu}(q, x), y)$$

for any state q and strings $x, y \in A^*$. Hint: use induction on $|y|$.

(b) Show that for any $q \in S, x \in A^*, a \in A$ we have

$$\widehat{\mu}(q, ax) = \widehat{\mu}(\mu(q, a), x).$$

Hint: use part (a).

Solution. (a) We use induction on $|y|$. The base case $|y| = 1$ is just the definition of $\widehat{\mu}$. Assume the statement is true for a given $y_0 \in A^*$. Then we must prove it for $y = y_0 a$ for any $a \in A$. (One could intepret this as a proof by structural induction over $y \in A^*$ rather than induction on $|y|$.)

We have

$$\begin{aligned}
\widehat{\mu}(q, xy) &= \widehat{\mu}(q, xy_0a) \\
&= \mu(\widehat{\mu}(q, xy_0), a) && \text{definition} \\
&= \mu(\widehat{\mu}(\widehat{\mu}(q, x), y_0), a) && \text{induction hypothesis} \\
&= \widehat{\mu}(\widehat{\mu}(q, x), y_0a) && \text{induction hypothesis} \\
&= \widehat{\mu}(\widehat{\mu}(q, x), y),
\end{aligned}$$

as required. Note that we had to use the induction hypothesis twice.

(b) We have

$$\begin{aligned}
\widehat{\mu}(q, ax) &= \widehat{\mu}(\widehat{\mu}(q, a), x) && \text{by (a)} \\
&= \widehat{\mu}(\mu(q, a), x) && \text{by definition,}
\end{aligned}$$

as required.

3. Write a regular expression for the following languages.

(a) The set of strings over the alphabet $\{a, b, c\}$ containing at least one a and at least one b.

(b) The set of strings of 0's and 1's whose third symbol from the right end is a 1.

Solution. (a) Let r be the regular expression $(\varepsilon + a + b + c)$. Then the regular expression we are looking for is

$$r^*ar^*br^* + r^*br^*ar^*.$$

(b) The sought regular expression is

$$(\varepsilon + 0 + 1)^*1(0 + 1)(0 + 1).$$

4. Let $\mathscr{L}$ be the set of strings of balanced parentheses. Thus $\mathscr{L}$ consists of the strings of characters "(" and ")" that can appear in a well-formed arithmetic expression. Show that $\mathscr{L}$ is not a regular language.

Solution. We use the pumping lemma. Suppose for contradiction that $\mathscr{L}$ is regular. Let n be such that given $w \in \mathscr{L}$ of length $\geqslant n$ there exists x, y, z with $y \neq \varepsilon$ and $|xy| \leqslant n$ such that $w = xyz$ and $xy^i z \in L$ for all i.

Define

$$w_j := \underbrace{((\cdots(}_{j}\underbrace{)\cdots))}_{j}$$

and consider w_N for some $N > n$. Then there exists x, y, z as in the pumping lemma. But since $|xy| \leqslant n < N$ we must have that y consists entirely of left parentheses. Thus for $i \geqslant 2$, $xy^i z$ will have more left parentheses than right parentheses and therefore cannot be a balanced string. This contradiction shows that $\mathscr{L}$ is not a regular language.

5. Let G be a group.

(a) If $N \triangleleft G$ is a normal subgroup, show that

$$\{(ng, g) : n \in N, g \in G\}$$

is a subgroup of $G \times G$.

(b) Show that the construction in part (a) establishes a bijection between the set of normal subgroups of G and the set of subgroups of $G \times G$ which contain the diagonal subgroup $\Delta := \{(g, g) \in G \times G : g \in G\}$.

Solution. (a) Let $H_N := \{(ng, g) : n \in N, g \in G\}$. To be a subgroup of $G \times G$ we must check that H_N has identity, multiplication, and inverse coming from $G \times G$. To see that $(1, 1) \in H_N$ just take $n = 1 \in N \subset G, g = 1 \in G$. To see that H_N is closed under multiplication we compute for $u, v \in N, g, h \in G$

$$(ug, g)(vh, h) = (ugvh, gh) = (u(gvg^{-1})gh, gh) \in H_N$$

since N is normal in G so that $gvg^{-1} \in N$. To see that H_N is closed under inversion we compute

$$(ng, g)^{-1} = ((ng)^{-1}, g^{-1}) = (g^{-1}n^{-1}, g^{-1}) = ((g^{-1}n^{-1}g)g^{-1}, g^{-1}) \in H_N$$

as before.

(b) Let $S := \{\text{normal subgroups } N \triangleleft G\}$ and $T := \{\text{subgroups } H \subset G \times G \text{ s.t. } H \supset \Delta\}$. We define a map $\alpha : S \to T$ by $\alpha(N) = H_N$ where $H_N := \{(ng, g) : n \in N, g \in G\}$ as in part (a), and we define $\beta : T \to S$ by $\beta(H) := N_H := \{ab^{-1} : (a, b) \in H\}$.

We must show that $\alpha \circ \beta = \mathrm{id}_T$ and $\beta \circ \alpha = \mathrm{id}_S$. This is the definition of a bijective correspondence. We compute

$$\beta(\alpha(N)) = \beta(H_N) = \beta(\{(ng, g) : n \in N, g \in G\}) = \{ngg^{-1} : n \in N, g \in G\} = N$$

and therefore $\beta \circ \alpha = \mathrm{id}_S$. Next, we have $\alpha(\beta(H)) = \alpha(N_H) = \{(ab^{-1}g, g) : (a, b) \in H, g \in G\}$. Therefore we must show that under the assumption $H \supset \Delta$ we have

$$\{(ab^{-1}g, g) : (a, b) \in H, g \in G\} = \{(a, b) \in H\}.$$

To prove $\supset$ it suffices to take $g = b$ in the left hand side. To prove $\subset$ we use the fact that $\Delta \subset H$. We then have

$$(ab^{-1}g, g) = (ab^{-1}, 1)(g, g) = (a, b)(b^{-1}, b^{-1})(g, g) \in H \cdot \Delta \cdot \Delta \subset H$$

as required.

6. Construct a deterministic automaton which accepts the following language over $\{0, 1\}$:

$$\mathscr{L} := \{x \in \{0, 1\}^* : x \text{ represents a multiple of 3 in binary}\}.$$

Leading zeros are permitted, and ε represents the number 0. For example, the string 001001 represents the number 0+0+8+0+0+1=9 and thus $001001 \in \mathscr{L}$.

Solution. To determine whether a number is a multiple of 3, we must compute the residue class of the number mod 3. Thus given a string we must decide which residue class mod 3 it lands in and accept the string iff it lands in the residue class of 0 mod 3. Therefore our set of states is the set of congruence classes mod 3: $S := \{q(0), q(1), q(2)\}$ where $q(i)$ indicates the class of $i \pmod 3$. The start state is $q(0)$ since the empty string corresponds to the number 0 and therefore the class of 0 mod 3. The set of accept states is $\{q(0)\}$. We claim that the transition matrix of the sought automaton is

$$\begin{array}{c|cc} & 0 & 1 \\ \hline q(0) & q(0) & q(1) \\ q(1) & q(2) & q(0) \\ q(2) & q(1) & q(2). \end{array}$$

We prove this claim by induction on the length of input string. Define $\text{eval} : \{0,1\}^* \to \mathbb{Z}$ by

$$\text{eval}(a_n a_{n-1} \cdots a_1 a_0) := \sum_{i=0}^{n} a_i 2^i$$

and define $\text{eval}(\varepsilon) := 0$. Thus $\text{eval}(100) = 4 + 0 + 0 = 4, \text{eval}(0010) = 0 + 0 + 2 + 0 = 2$, etc. Then our claim is equivalent to the claim that

$$\widehat{\mu}(q(0), w) = q(\text{eval}(w) \pmod 3). \tag{$*$}$$

In other words, the left hand side of $(*)$ is defined by means of the above transition matrix, and we must show that this agrees with the right hand side of $(*)$ for each $w \in \{0,1\}^*$.

We prove $(*)$ by induction on the length of w. The base case $|w| = 1$ follows from inspecting the above table and comparing with the definition of eval. Assume the induction hypothesis that $(*)$ is true for w. We must prove it is true for the string wa where $a \in \{0,1\}$. (As so often happens, one could say that we are doing a structural induction over all strings instead of a classical induction on the length of the strings; they amount to the same thing in this proof.)

We have

$$\text{eval}(wa) = 2(\text{eval}(w)) + a \tag{$**$}$$

where on the left hand side we understand the a to mean an element 0 or 1 of the alphabet, and on the right hand side we understand a to be the number 0 or $1 \in \mathbb{Z}$. Thus we have

$$\begin{aligned} \widehat{\mu}(q(0), wa) &= \mu(\widehat{\mu}(q(0), w), a) & \\ &= \mu(q(\text{eval}(w) \pmod 3), a) & \text{by the induction hypothesis} \\ &= q(2(\text{eval}(w)) + a \pmod 3) & \text{by the definition of } q \\ &= q(\text{eval}(wa) \pmod 3) & \text{by } (**), \end{aligned}$$

as required.