ETH zürich
Departement Informatik

Theoretische Informatik

Prof. Dr. Juraj Hromkovič
Dr. Hans-Joachim Böckenhauer
https://courses.ite.inf.ethz.ch/theo_inf_21

Exemplary Solutions – Sheet 4

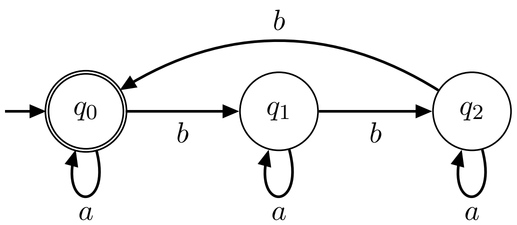
Zürich, October 22, 2021

Solution to Exercise 10

The language L from the task statement can be expressed as $L = L_1 \cup L_2$ where

$$\begin{aligned} L_1 &= \{w \in \{a,b\}^* \mid |w|_a \bmod 3 = |w| \bmod 3\} \\ &= \{w \in \{a,b\}^* \mid |w|_a \bmod 3 = (|w|_a + |w|_b) \bmod 3\} \\ &= \{w \in \{a,b\}^* \mid |w|_b \bmod 3 = 0\} \\ L_2 &= \{w \in \{a,b\}^* \mid w \text{ contains the subword } ab \text{ and } w \text{ ends by } b\}\,. \end{aligned}$$

The following automaton A_1 can be constructed for the language L_1:

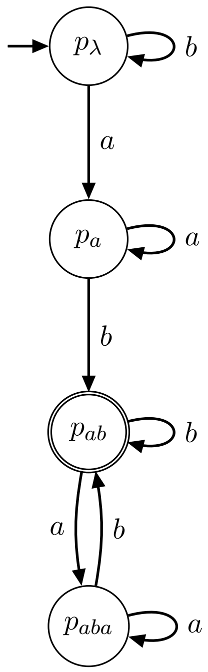


This automaton evaluates the expression $|w|_b$ modulo 3 in its states. It has the following classes, for $i \in \{0,1,2\}$:

$$\text{Kl}[q_i] = \{w \in \{a,b\}^* \mid |w|_b \bmod 3 = i\}\,.$$

(please turn over)

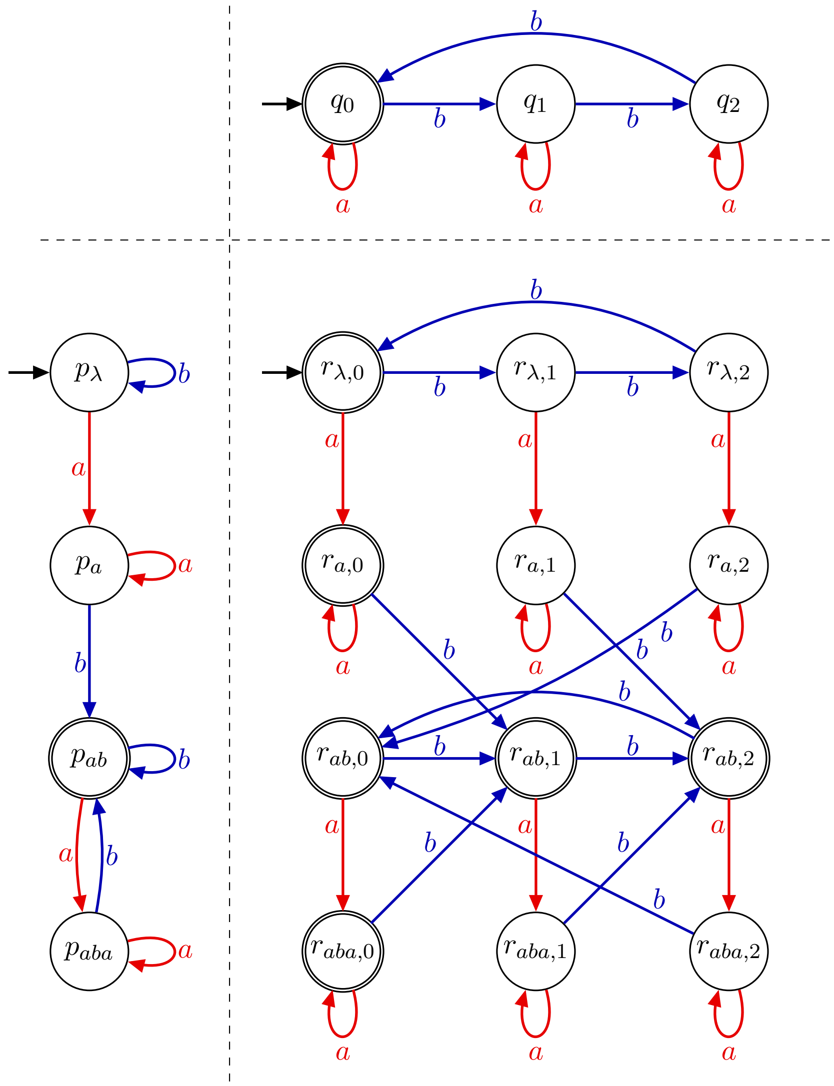
The following automaton A_2 can be constructed for the language L_2:



The states p_z, for $z \in \{\lambda, a, ab\}$, represent the longest prefix of the pattern ab read so far, p_{ab} is the accepting state. The state p_{aba} is reached if ab has already been read and a is the last symbol read so far. Hence, the classes are:

$$\begin{aligned}
\mathrm{Kl}[p_\lambda] &= \{b\}^*\,, \\
\mathrm{Kl}[p_a] &= \{xay \mid x \in \{b\}^* \text{ and } y \in \{a\}^*\}\,, \\
\mathrm{Kl}[p_{ab}] &= L_2\,, \\
\mathrm{Kl}[p_{aba}] &= \{a, b\}^* - \bigcup_{z \in \{\lambda, a, ab\}} \mathrm{Kl}[p_z]\,.
\end{aligned}$$

Applying the product automaton construction to A_1 and A_2, we can now construct the following product automaton A that accepts the language $L = L_1 \cup L_2$. For the sake of its representation, we use the notation $r_{x,y} = \langle p_x, q_y \rangle$. Because our product automaton is supposed to accept the union of two languages, a state of the product automaton is accepting if and only if it contains an accepting state of one of the two subautomata, i.e., the column corresponding to q_0 and the row corresponding to p_{ab} consist of accepting states of the product automaton.



Solution to Exercise 11

To prove that every finite automaton accepting the language L contains at least 5 states, we determine 5 words $w_1, \ldots, w_5$ and show that the automaton must reach 5 pairwise distinct states on these words. If there is a finite automaton that reaches the same state when reading two distinct words w_i and w_j, then Lemma 3.12 from the textbook implies that, for every $z \in \Sigma^*$, the automaton also reaches the same state when reading the words $w_i z$ and $w_j z$. We show that this is not possible for an automaton with less than 5 states by providing, for every two distinct words w_i and w_j, a word $z_{i,j}$ such that

$$w_i z_{i,j} \in L(A) \iff w_j z_{i,j} \notin L(A)\,. \qquad (1)$$

We choose the five words $w_1 = \lambda$, $w_2 = a$, $w_3 = b$, $w_4 = ab$, and $w_5 = ba$.

(please turn over)

The following table provides, for all pairs (w_i, w_j) with $i < j$, a word $z_{i,j}$.

$z_{i,j}$	$w_2 = a$	$w_3 = b$	$w_4 = ab$	$w_5 = ba$
$w_1 = \lambda$	b	a	λ	λ
$w_2 = a$	–	a	λ	λ
$w_3 = b$	–	–	λ	λ
$w_4 = ab$	–	–	–	a

It is easy to see that these words satisfy the condition (1), e.g., $w_4 z_{4,5} = aba \in L(A)$, but $w_5 z_{4,5} = baa \notin L(A)$.

Solution to Exercise 12

(a) We use Lemma 3.12 from the textbook to show that the language

$$L_1 = \{w \in \{a, b, c\}^* \mid w \text{ contains the subword } ab \text{ as many times as the subword } ba\}$$

is not regular. Suppose that L_1 is regular. Then there exists an automaton $A = (Q, \{a, b, c\}, \delta, q_0, F)$ with $L(A) = L_1$. Let $m = |Q|$. We consider the words

$$\lambda, abc, (abc)^2, \ldots, (abc)^m .$$

Because these are $m + 1$ words, i.e., more words than the number of A's states, there exist $i, j \in \{0, \ldots, m\}$ with $i \neq j$ such that

$$\hat{\delta}(q_0, (abc)^i) = \hat{\delta}(q_0, (abc)^j) .$$

Lemma 3.12 implies that, for all $z \in \{a, b, c\}^*$,

$$(abc)^i z \in L_1 \iff (abc)^j z \in L_1.$$

However, choosing $z = (bac)^i$ leads to a contradiction because

$$(abc)^i z = (abc)^i (bac)^i \in L_1$$

and $(abc)^j z = (abc)^j (bac)^i \notin L_1$. Hence, the assumption is wrong and L_1 is not regular.

(please turn over)

(b) We use the pumping lemma to show that the language

$$L_2 = \{w \in \{0,1\}^* \mid |w|_0 \neq |w|_1\}$$

is not regular. Suppose that L_2 is regular. Then the pumping lemma (Lemma 3.14) implies that there exists a constant $n_0 \in \mathbb{N}$ such that every word $w \in \{0,1\}^*$ with $|w| \geq n_0$ can be split into three parts y, x, and z such that

1. $|yx| \leq n_0$,
2. $|x| \geq 1$, and
3. either $\{yx^k z \mid k \in \mathbb{N}\} \subseteq L_2$ or $\{yx^k z \mid k \in \mathbb{N}\} \cap L_2 = \emptyset$.

We choose the word $w = 0^{n_0}1^{n_0}$. Then we clearly have $|w| \geq n_0$. Hence, there exists a decomposition $w = yxz$ of w satisfying the conditions (i), (ii), and (iii). The condition (i) implies $|yx| \leq n_0$, i.e., $y = 0^l$ and $x = 0^m$ for $l, m \in \mathbb{N}$ such that $l + m \leq n_0$, in particular, $m \leq n_0$. The condition (ii) further implies $m > 0$. Because $w \notin L_2$, the condition (iii) implies that

$$\{yx^k z \mid k \in \mathbb{N}\} = \{0^{n_0+(k-1)m}1^{n_0} \mid k \in \mathbb{N}\} \cap L_3 = \emptyset\,.$$

However, this is a contradiction because $yx^2z = 0^{n_0+m}1^{n_0} \in L_2$. Hence, the assumption is wrong and L_2 is not regular.

(c) We use the Kolmogorov complexity argument to show that the language

$$L = L_3 = \{0^{\binom{2n}{n}} \mid n \in \mathbb{N}\}$$

is not regular. Suppose that L_3 is regular. For every $n \geq 1$, we define

$$\Delta_n := \binom{2(n+1)}{n+1} - \binom{2n}{n} = \binom{2n}{n} \cdot \left(\frac{(2n+1)(2n+2)}{(n+1)^2} - 1\right) = \binom{2n}{n} \cdot \left(3 - \frac{2}{n+1}\right).$$

For every $m \in \mathbb{N}$, $0^{\Delta_m - 1}$ is thus the first word in the language

$$L_{0^{\binom{2m}{m}+1}} = \{y \mid 0^{\binom{2m}{m}+1} y \in L\}\,.$$

Theorem 3.19 from the textbook implies that there exists a constant c, independent of m, such that

$$K(0^{\Delta_m - 1}) \leq \lceil \log_2(1+1) \rceil + c = 1 + c\,.$$

Because there are only finitely many programs of constant length at most $1 + c$, but infinitely many words of the form $0^{\Delta_m - 1}$, this leads to a contradiction. Hence, the assumption is wrong and L_3 is not regular.