ETH zürich
Departement Informatik

Theoretische Informatik

Prof. Dr. Juraj Hromkovič
Dr. Hans-Joachim Böckenhauer
https://courses.ite.inf.ethz.ch/theoInf20

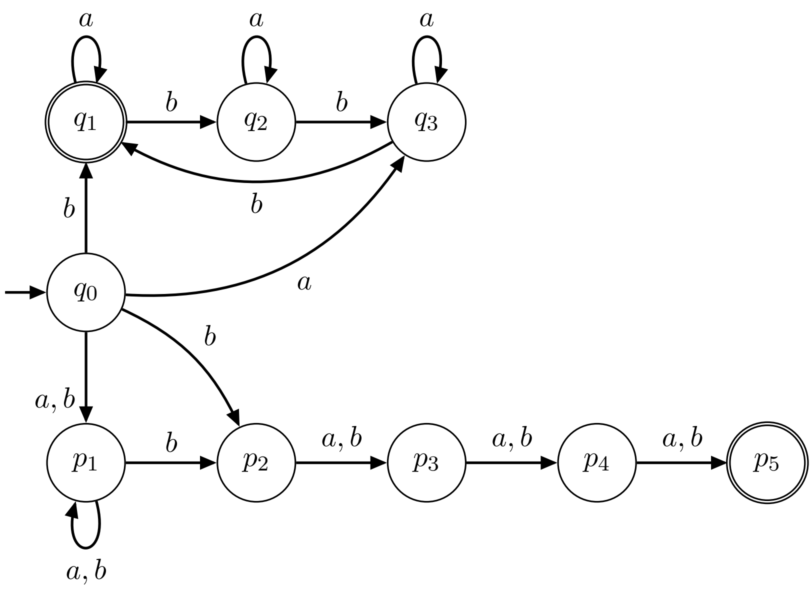
Exemplary Solutions – Sheet 6

Zürich, October 30, 2020

Solution to Exercise 15

(a) The following nondeterministic finite automaton M_1 accepts the language

$$L_1 = \{x \in \{a,b\}^* \mid |x|_b \bmod 3 = 1 \text{ or } (x = ybz \text{ with } y,z \in \{a,b\}^* \text{ and } |z| = 3)\}\,.$$

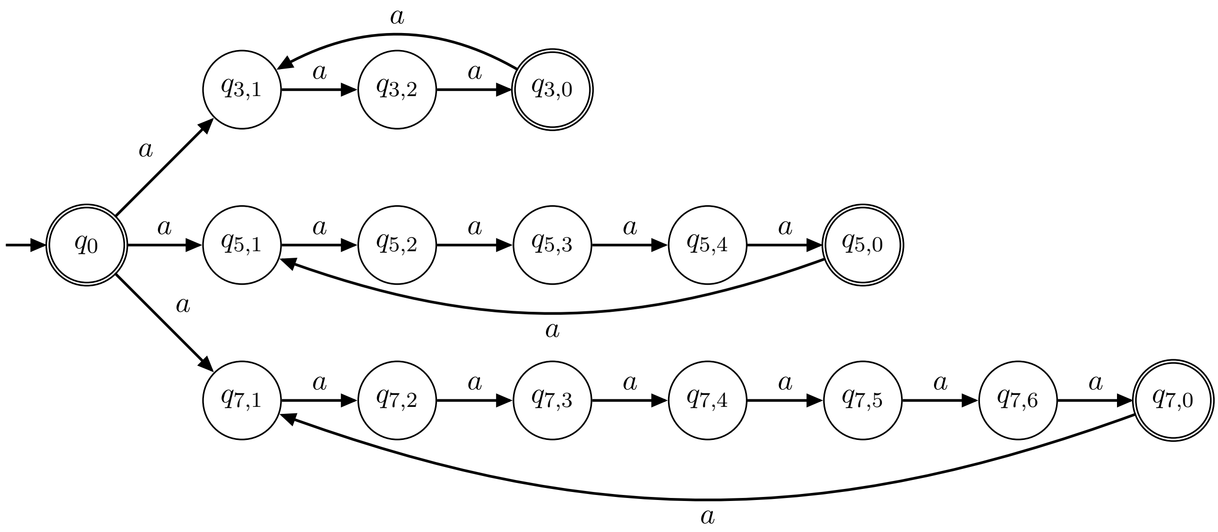


The NFA consists of two subautomata for the two conditions of the language L. From the initial state q_0, M_1 branches nondeterministically into one of the two subautomata. In the states q_1, q_2, and q_3, M_1 counts the number of b's in the input word modulo 3. If the count equals 1, then the first of the two conditions holds and M_1 accepts the input in the state q_1. In the states p_1 through p_5, M_1 checks whether the fourth-to-last symbol is a b, i.e., whether the second condition of L holds. To this end, M_1 decides nondeterministically in the state p_1 when the fourth-to-last symbol has been reached. Using the states p_2 up to p_5, M_1 reads exactly three arbitrary symbols on the way from p_2 into the state p_5. If the entire input has been read by then, M_1 accepts in the state p_5.

(b) The following nondeterministic finite automaton M_2 accepts the language

$$L_2 = \{a^n \mid n \in \mathbb{N} \text{ is divisible by } 3, 5, \text{ or } 7\}$$

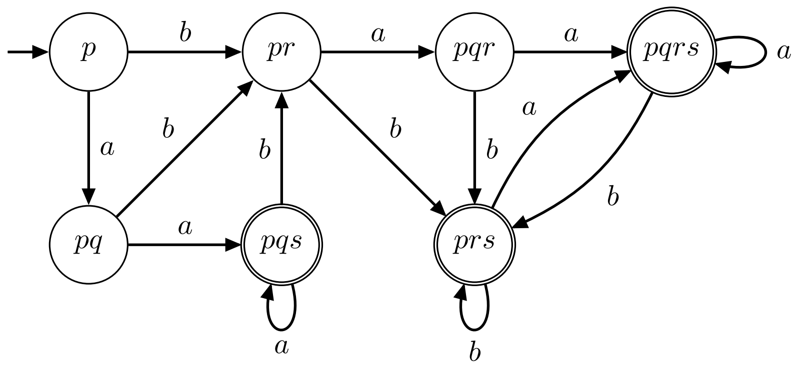
over the alphabet $\{a\}$.



The NFA consists of three subautomata for the three divisors 3, 5, and 7 in the definition of the language L. From the initial state q_0, M_2 branches nondeterministically into one of the three subautomata. In the states $q_{i,j}$, M_2 counts the number of a's in the input modulo i. If the count equals 0, then the corresponding divisibility condition holds and M_2 accepts the input in the state $q_{i,0}$. Moreover, M_2 accepts the empty input $\lambda = a^0$ in the state q_0, as 0 is divisible by all positive natural numbers.

Solution to Exercise 16

(a) Applying the powerset construction to the NFA from the exercise sheet yields the following deterministic finite automaton A. All nonreachable states have been dropped. For the sake of readability, the labels of the states have been shortened, e.g., pqr denotes the state $\langle\{p, q, r\}\rangle$.



(b) To show that every deterministic finite automaton that accepts the same language as the automaton A from part (a) requires at least 6 states, we fix 6 words $w_1, \ldots, w_6$

and show that the automaton reaches 6 pairwise distinct states upon reading them. If there exists a FA that reaches the same state upon reading two distinct words w_i and w_j, then, by Lemma 3.3 from the textbook, for every $z \in \{a,b\}^*$, the automaton reaches the same state upon reading the words $w_i z$ and $w_j z$, too. We seek to show that this is impossible for an automaton with less than 6 states. To this end, for every pair of distinct words w_i and w_j, we provide a word $z_{i,j}$ such that

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

To determine the words $w_1, \ldots, w_6$, we consider the FA A. It has 7 states, but we note that the two states $\langle\{p,q,r,s\}\rangle$ and $\langle\{p,r,s\}\rangle$ can be merged because they are both accepting and the automaton cannot leave the set of these two states. We thus choose, for each state except $\langle\{p,q,r,s\}\rangle$, a shortest word from its class as a candidate. These are the words $w_1 = \lambda$, $w_2 = a$, $w_3 = b$, $w_4 = aa$, $w_5 = ba$, and $w_6 = bb$.

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

$z_{i,j}$	$w_2 = a$	$w_3 = b$	$w_4 = aa$	$w_5 = ba$	$w_6 = bb$
$w_1 = \lambda$	a	b	λ	b	λ
$w_2 = a$	–	a	λ	b	λ
$w_3 = b$	–	–	λ	a	λ
$w_4 = aa$	–	–	–	λ	b
$w_5 = ba$	–	–	–	–	λ

It is easy to see that these words satisfy the condition (1), e.g., $w_3 z_{3,5} = ba \notin L(A)$, but $w_5 z_{3,5} = baa \in L(A)$.