Theoretical Computer Science 1
Exercise Sheet 5

René Maseli
Prof. Dr. Roland Meyer
TU Braunschweig
Winter semester 2022/23

Release: 10.01.2023 Due: 20.01.2023, 09:45

Hand in your solutions to the Vips directory of the StudIP course until Friday, 20.01.2022 **09:45** pm. You should provide your solutions either directly as .pdf file or as a readable scan/photo of your handwritten notes. Submit your results as a group of four.

Exercise 1: Equivalence classes [10 points]

Let $\Sigma = \{a, b\}$ be an alphabet.

a) [4 points] Consider $L = \{a^n b^m \mid n, m \in \mathbb{N}, n \geq m\}$. Prove that

$$[a^n]_{\equiv_L} = \{a^n\} \text{ for all } n \in \mathbb{N}$$
$$[a^n.a.b]_{\equiv_L} = \left\{a^{\ell+1}.b^{\ell+1-n} \,\middle|\, \ell \in \mathbb{N}, \ell \geqslant n\right\} \text{ for all } n \in \mathbb{N}$$

holds.

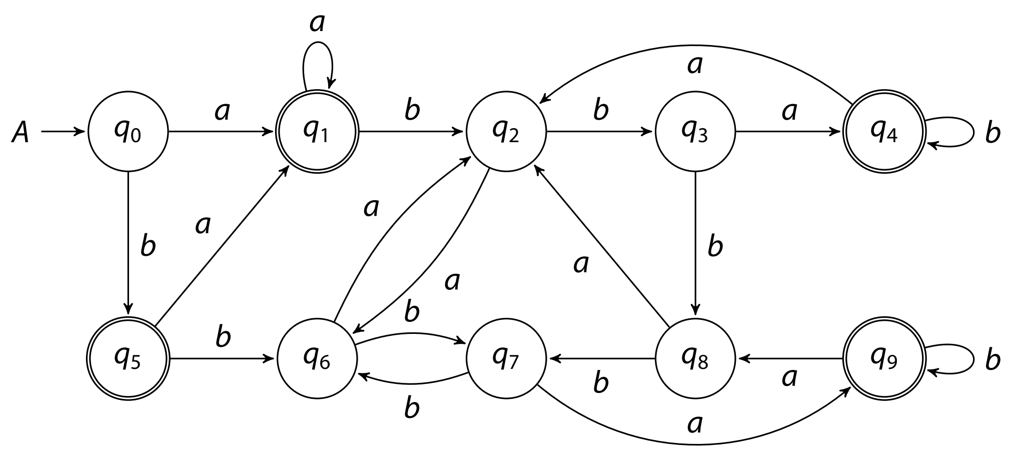
Find all remaining equivalence classes with respect to $\equiv_L$. In particular, for all $n, m \in \mathbb{N}$ determine the equivalence class of $a^n b^m$. (You do not have to give a formal proof.)

b) [3 points] Consider the language $M = \{a, b\}^*.\{aab, abb\}.\{a, b\}^*$. Find all equivalence classes of $\equiv_M$. Construct the equivalence class automaton A_M.

c) [3 points] Consider the language $N = \{a, b\}^*.\{a\}.\{a, b\}^* \cup (\{a, b\}.\{a, b\})^*$. Find all equivalence classes of $\equiv_N$. Construct the equivalence class automaton A_N.

Exercise 2: Minimization [10 points]

Consider the following NFA A over $\{a, b\}$.



a) [5 points] Determine the ~-equivalence classes on the states of A by using the Table-Filling-Algorithm from the lecture. Make clear in which order the cells of the table were marked.

b) [2 points] Give the minimal DFA B for $\mathcal{L}(A)$. Make use of the ~-equivalence classes.

c) [3 points] Find all equivalence classes of the Nerode right-congruence $\equiv_{\mathcal{L}(A)}$. Find an expression for $\mathcal{L}(A)$ as a union of a certain subset of those classes.

Exercise 3: Pumping Lemma [6 points]

Consider $\Sigma = \{a, b\}$. For any word w let $|w|_a$ be the number of occurrences of symbol a in w. $|w|_b$ is defined analogously.

By using the Pumping Lemma, prove that the following languages are not regular.

a) [2 points] $L_1 = \{w \in \{a, b\}^* \mid |w|_b + 7 > |w|_a\}$

b) [4 points] $L_2 = \{(ab)^n b^m w \in \{a, b\}^* \mid |w|_a = n, m \geq 2\}$

Exercise 4: Context free grammars [9 points]

Consider $\Sigma = \{a, b\}$. Give context free grammars G_1, G_2 and G_3, which produce the following languages:

a) [1 point] $\mathcal{L}(G_1) = \{a^n b^m w \mid w \in \Sigma^*, m > 2, |w|_a = n\}$.

b) [2 points] $\mathcal{L}(G_2) = \{w \in \Sigma^* \mid |w|_a < |w|_b\}$.

c) [2 points] $\mathcal{L}(G_3) = \{w \in \Sigma^* \mid \forall u, v\colon\ w = u.v \Rightarrow |v|_a \leq |v|_b\}$.

A context free grammar G is called **regular** if it is left linear or right linear. Right linear means that the right-hand sides of all production rules contain at most one non-terminal which (if it exists) is at the right most position. Hence, all rules are of the form $X \to w$ or $X \to w.Y$ where $w \in \Sigma^*$. Left linear is defined similarly.

Prove that the regular languages exactly coincide with the languages that are produced by some right linear grammar G.

d) [2 points] Explain how to construct a right linear grammar G from a given NFA A such that $\mathcal{L}(G) = \mathcal{L}(A)$ holds.

e) [2 points] Explain how to construct an NFA A from a given right linear grammar G such that $\mathcal{L}(G) = \mathcal{L}(A)$ holds.

Remark: An analogous result holds for left linear grammars as well. That is why we speak of **regular** grammars in both cases.