Theoretical Computer Science 1
Exercise Sheet 5

Thomas Haas
Prof. Dr. Roland Meyer

TU Braunschweig
Winter semester 2019/20

Release: 07.01.2020 Due: 16.01.2020, 15:00

Hand in your solutions by Thursday 3pm, 2020/01/16, by inserting them into the exercise boxes next to office IZ 343. Please hand in in groups of 4 people.

Exercise 1: Equivalence classes [7 points]

a) [4 points] Consider the language

$$\mathcal{L} = \left\{a^n b^n \;\middle|\; n \in \mathbb{N}\right\}.$$

Prove that

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

holds.

Find all remaining equivalence classes with respect to $\equiv_\mathcal{L}$. 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

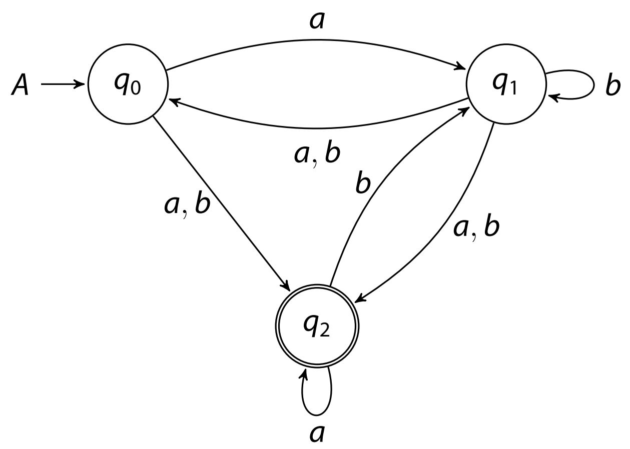
$$\mathcal{L} = \left\{w \in \{a, b\}^* \;\middle|\; w \text{ contains } aa \text{ or } bb\right\}.$$

Find all equivalence classes of $\equiv_\mathcal{L}$.
Construct the equivalence class automaton $A_\mathcal{L}$.

Hint: With "contains *aa*" we mean that w is of the form $w = w_1.a.a.w_2$.

Exercise 2: Minimization [8 points]

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



a) [2 points] From A, construct a language equivalent DFA B using the Rabin-Scott power set construction.

   Make sure that B has no unreachable states.

b) [3 points] Determine the ~-equivalence classes on the states of B by using the Table-Filling-Algorithm from the lecture.

   Make clear in which order the cells of the table were marked.

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

d) [1 point] Compare the size (number of states) of A, B and C.

Exercise 3: Pumping Lemma [6 points]

a) [3 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

$$\mathcal{L} = \left\{ w \in \Sigma^* \,\middle|\, |w|_b + 7 > |w|_a \right\}$$

   is not regular.

b) [3 points] By using the Pumping Lemma, prove that

$$\mathcal{L} = \left\{ w \in \Sigma^* \,\middle|\, |w|_a \neq |w|_b \right\}$$

   is not regular.

   Hint: Think about the following: For some natural number $n \in \mathbb{N}$, which number is divisible by all numbers $\leqslant n$?

Exercise 4: Context free grammars [10 points]

a) For each of the following languages, give a context free grammar that produces that language.

   i) [2 points] $\mathcal{L}_1 = \{w \in \{a, b, (,)\}^* \mid w \text{ is correctly parenthesized}\}$

   ii) [2 points] $\mathcal{L}_2 = \{w \in \{a, b\}^* \mid |w|_a = |w|_b\}$

   iii) [2 points] $\mathcal{L}_3 = \{w \in \{a, b\}^* \mid |w|_a \neq |w|_b\}$

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.

   - [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.
   - [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.