

Theory of Computer Science

G. Röger
Spring Term 2020

University of Basel
Computer Science

Exercise Sheet 5

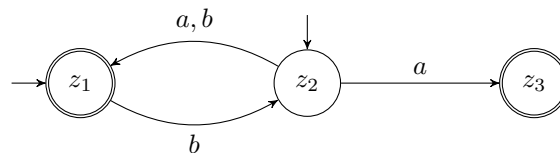
Due: Wednesday, April 1, 2020

Exercise 5.1 (DFA and NFA, 1.5+1.5 marks)

- (a) Specify a deterministic finite automaton that accepts the language of all words over $\Sigma = \{a, b\}$ that do *not* contain **bab** (e.g., the word **ababa** is not contained).
- (b) Specify a non-deterministic finite automaton that accepts the language of those words over $\Sigma = \{a, b\}$ that start with **ab** or contain **abba**.

Exercise 5.2 (DFA and NFA, 2 marks)

Specify a DFA that is equivalent to the following NFA.



Exercise 5.3 (Regular Expressions, 2 marks)

Consider the following regular expressions over the alphabet $\Sigma = \{0, 1\}$. For each regular expression, specify two words that are in the corresponding language and two words that are not in the corresponding language.

- (a) $110|1001$
- (b) $1^*(01^*01^*)^*$
- (c) $(0\varepsilon|1(0|1))(0|1)^*$
- (d) $1(\varepsilon|0)|001$

Exercise 5.4 (Regular Expressions, 1 mark)

Specify a regular expression that describes the language

$$L = \{w \in \{0, 1\}^* \mid |w| \geq 2, w \text{ ends with } 0 \text{ and contains at most two } 0\text{s}\}.$$

Exercise 5.5 (NFAs for Regular Expressions; 2 Points)

Construct an NFA for the regular expression $((ab)^*|a^*)$ over the alphabet $\Sigma = \{a, b\}$. Use the construction rules from the lecture (chapter C3 slides 13–16 on handout version) and please specify all intermediate steps, i.e., NFAs for **a**, **b**, **ab**, $(ab)^*$, a^* and $((ab)^*|a^*)$.