General notes

Conventions used throughout this exam

- $\Sigma_{\text{bin}} = \{\texttt{0}, \texttt{1}\}$.
- $\Sigma_{\text{abc}} = \{\texttt{a}, \texttt{b}, \texttt{c}\}$.
- Σ_{ASCII} is the ASCII alphabet.
- $\Sigma_{\text{cap}} \subset \Sigma_{\text{ASCII}}$ is the set of all capitals.
- $\Sigma_{\text{dig}} \subset \Sigma_{\text{ASCII}}$ is the set of all decimal digits.
- "Formally describe a language" means that a formal language is to be described in a set-theoretic way, e.g.

$$L = \{\, w \in \Sigma_{\text{bin}}^* \,|\, |w| < 5\}$$

- *Binary numbers* are arbitrary words of Σ_{bin}^*, however, they must not begin with `0` unless the word is `0` itself.
- If not noted otherwise, regular expressions are based on the operator set $\{+, \cdot, ^*, (,)\}$.

Algebraic operations on regular expressions

1. $r_1 + r_2 \doteq r_2 + r_1$ (commutative law)
2. $(r_1 + r_2) + r_3 \doteq r_1 + (r_2 + r_3)$ (associative law)
3. $(r_1 r_2) r_3 \doteq r_1 (r_2 r_3)$ (associative law)
4. $\emptyset r \doteq \emptyset$
5. $\varepsilon r \doteq r$
6. $\emptyset + r \doteq r$
7. $(r_1 + r_2) r_3 \doteq r_1 r_3 + r_2 r_3$ (distributive law)
8. $r_1 (r_2 + r_3) \doteq r_1 r_2 + r_1 r_3$ (distributive law)
9. $r + r \doteq r$
10. $(r^*)^* \doteq r^*$
11. $\emptyset^* \doteq \varepsilon$
12. $\varepsilon^* \doteq \varepsilon$
13. $r^* \doteq \varepsilon + r^* r$

14. $r^* \doteq (\varepsilon + r)^*$

15. $\{r \doteq rs + t \quad \text{with} \quad \varepsilon \notin L(s)\} \longrightarrow r \doteq ts^*$ (proof by Arto Salomaa)

58. $r^* r \doteq r r^*$

65. $r^* \doteq \varepsilon + r^*$

1 Formal languages (6 pts)

Formally describe the following languages:

a) All odd binary numbers.

b) All binary numbers whose division by `100` is even.

c) A typical German license plate (consisting of one to three capitals, followed by a space, followed by one or two capitals, followed by a space, followed by one to four digits).

2 Regular expressions (8 pts)

a) Give a regular expression for all the words $w \in \Sigma_{\text{abc}}^*$ containing at least one of each character in Σ_{abc}.

b) Give a regular expression for all the words $w \in \Sigma_{\text{abc}}^*$ *not* containing the substring `ab`.

c) Determine the shortest word w_s not matched by

$$r = (\mathtt{1}^*\mathtt{01})^*\mathtt{1}^*\mathtt{0}^*.$$

Formally describe the language $L(r)$ making use of w_s.

3 Algebraic operations (5 pts)

For this section, use the 15 algebraic operations on regular expressions introduced in the lecture. You can also use Equations (58) and (65) as we have proven them before. For the sake of simplicity, the rules are listed in the General Notes section above.

a) Simplify as much as possible:

$$r = (\varepsilon + \emptyset)^*(\varepsilon + t)^*(s + \emptyset t) + (\varepsilon + t^*)^*(\varepsilon + t^*)s + s$$

b) Prove that

$$\mathtt{1}(\mathtt{001})^* \doteq (\mathtt{100})^*\mathtt{1}.$$

In subtask b), indicate which rules you are using by referring to their ID (1-15, 58, 65) at every step you take.

4 JFlex (5 pts)

Give regular expressions using the JFlex operator set matching

a) The language described in Task 1 c).

b) The regular expression described in Task 2 b).

c) The first sentence of a text.
For the sake of simplicity, assume that

- sentences are separated by periods which are not used for any other purpose,
- words are separated by single empty spaces, and,
- otherwise, the text only contains alphanumeric characters and
- has no leading spaces.

5 Transforming FSMs into regular expressions (5 pts)

Determine a regular expression representing the same language as the following NFA:

$$A = \langle Q, \Sigma, \delta, q_0, F \rangle$$

with

1. $Q = \{0, 1, 2\}$
2. $\Sigma = \Sigma_{\text{bin}}$
3. $\delta = \{\langle 0, \varepsilon, 0 \rangle \, \langle 0, \mathtt{0}, 1 \rangle, \langle 0, \mathtt{0}, 2 \rangle, \langle 0, \mathtt{1}, 1 \rangle, \langle 0, \mathtt{1}, 2 \rangle, \langle 1, \varepsilon, 0 \rangle,$
$\langle 1, \varepsilon, 2 \rangle, \langle 1, \mathtt{1}, 0 \rangle, \langle 1, \mathtt{1}, 1 \rangle, \langle 2, \varepsilon, 0 \rangle, \langle 2, \varepsilon, 1 \rangle, \langle 2, \mathtt{1}, 2 \rangle\}$
4. $q_0 = 0$
5. $F = \{2\}$

6 Transforming regular expressions into FSMs (8 pts)

Transform the regular expression given in Task 2 c) into a minimized DFA.

7 Formal grammars (8 pts)

a) Define a formal grammar representing the language specified in Task 1 c). Note: Use quotes to represent terminal symbols (e.g. “A”, “1”, “ ”).

b) Consider the grammar

$$G = \langle V_N, V_T, P, S\rangle$$

with

1. $V_N = \{S, A\}$,
2. $V_T = \{\texttt{0}, \texttt{1}\}$,
3. P :

$$\begin{aligned} S &\rightarrow \texttt{00}A & 1 \\ S &\rightarrow \texttt{11}A & 2 \\ A &\rightarrow \texttt{01}A & 3 \\ A &\rightarrow \texttt{10}A & 4 \\ A &\rightarrow \varepsilon & 5 \end{aligned}$$

4. $S = S$.

i. What is G's highest type? Justify your answer.

ii. Formally describe the language $L(G)$.

iii. If G is not itself regular, define a regular grammar G' with

$$L(G') = L(G).$$

8 The pumping lemma (2 pts)

Considering the language $L(G)$ given in Task 7 b) ii, which of the following statements is true:

a) Application of the pumping lemma proves that $L(G)$ is regular.

b) Application of the pumping lemma disproves that $L(G)$ is regular.

c) Application of the pumping lemma neither proves nor disproves that $L(G)$ is regular.

Justify your answer. Note: Keep your answer brief; no need for lots of math.