

## Automata and Formal Languages — Endterm

- You have 120 minutes to complete the exam.
- Answers must be written in a separate booklet. Do not answer on the exam.
- Please let us know if you need more paper.
- Write your name and Matrikelnummer on every sheet.
- Write with a non-erasable pen. Do not use red or green.
- You are not allowed to use auxiliary means other than pen and paper.
- You can obtain 40 points. You need 17 points to pass.

### Question 1 (6 points)

- (a) Give the minimal DFA for the language  $\{w \in \{a, b\}^* \mid w \text{ does not contain } bba\}$ .
- (b) Prove or disprove: if  $L_1$  and  $L_1 \cdot L_2$  are regular languages, then  $L_2$  is regular. (To prove it, show how to construct an NFA for  $L_2$  from NFAs for  $L_1$  and  $L_1 \cdot L_2$ . To disprove it, give languages  $L_1$  and  $L_2$  such that  $L_1$  and  $L_1 \cdot L_2$  are regular, and  $L_2$  is not regular.)
- (c) Give a transducer over alphabet  $\{0, 1\} \times \{0, 1\}$  accepting least significant bit first (lsbf) encodings of pairs  $(x, y) \in \mathbb{N} \times \mathbb{N}$  such that  $x > 0$  and  $y = x - 1$ .
- (d) Let  $\varphi = \mathbf{GF}(\neg p) \vee \mathbf{FG}p \vee (\neg p \mathbf{U} (\mathbf{XX}p))$  be an LTL formula over the set of atomic propositions  $AP = \{p\}$ . Give an  $\omega$ -regular expression  $s$  such that  $L_\omega(\varphi) = L_\omega(s)$ .
- (e) Give an  $\omega$ -word satisfying  $((\mathbf{X}p) \mathbf{U} q) \wedge ((\mathbf{X}\neg p) \mathbf{U} (\neg q))$ .
- (f) Give a Muller automaton over the alphabet  $\Sigma = \{a, b, c\}$ , with at most 3 states, recognizing the language of all  $\omega$ -words containing infinitely many  $a$ 's, infinitely many  $b$ 's and infinitely many  $c$ 's.

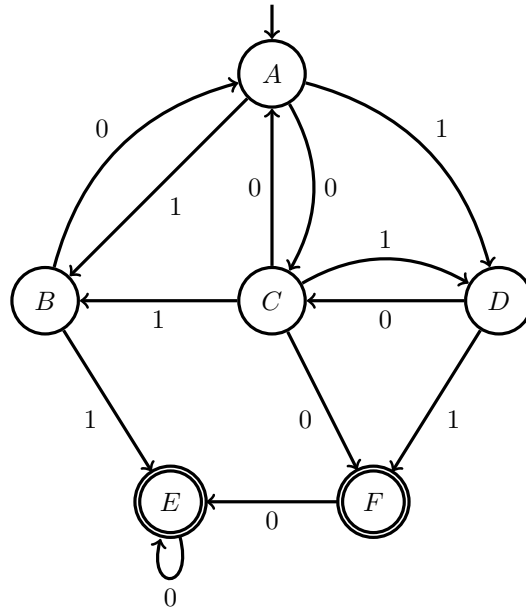
### Question 2 (2 + 2 + 2 = 6 points)

Decide whether the following languages over alphabet  $\{a, b\}$  are regular or not. If a language is regular, give a corresponding automaton or regular expression. If a language is not regular, prove that it has infinitely many residuals.

- (a)  $L_1 = \{w \in \{a, b\}^* \mid w \text{ contains as many } a\text{'s as } b\text{'s}\}$ .
- (b)  $L_2 = \{a^m b^n \mid m \geq n\}$ .
- (c)  $L_3 = \{a^m b^n \mid m < 1000 \wedge m \leq n\}$ .

**Question 3 (1 + 2 + 2 = 5 points)**

Reduce the size of the following NFA by means of the partition refinement algorithm introduced in the course.



- Give the blocks of the initial partition.
- Describe for each step which block is being refined, which is the splitter, and which are the two blocks resulting from the split.
- Draw the quotient of the original automaton with respect to the partition computed in (b).

**Question 4 (3 + 3 = 6 points)**

Recall that, given a state  $q$  of the master automaton for fixed-length languages over an alphabet  $\Sigma$ , and given  $a \in \Sigma$ , we denote  $q^a$  the unique  $a$ -successor of  $q$ . Further, we denote by  $q_\epsilon$  and  $q_\emptyset$  the states of the master automaton recognizing  $\{\epsilon\}$  and  $\emptyset$ , respectively.

- (a) Fill the blanks in the algorithm for computing the state of the master automaton recognizing the intersection of the languages recognized by two given states. The alphabet is  $\Sigma = \{a_1, a_2, \dots, a_m\}$ .

**Input:** states  $q_1, q_2$  with same length.

**Output:** state recognizing  $L(q_1) \cap L(q_2)$ .

**inter**( $q_1, q_2$ ):

**if**  $G(q_1, q_2)$  is not empty **then return**  $G(q_1, q_2)$

**else if** Blank 1 **then return**  $q_\emptyset$

**else if** Blank 2 **then return**  $q_\epsilon$

**else**

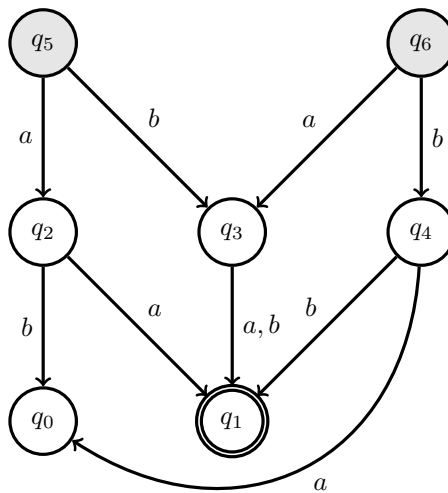
**for**  $i = 1, \dots, m$  **do**

$r_i \leftarrow$  Blank 3

$G(q_1, q_2) \leftarrow \text{make}(r_1, \dots, r_m)$

**return**  $G(q_1, q_2)$

- (b) Apply the algorithm to compute the state recognizing  $L(q_5) \cap L(q_6)$  in the fragment of the master automaton shown below. Describe the tree of recursive calls, give the result of each call, and draw the resulting automaton. If you use optimizations described in the course, then describe them briefly.



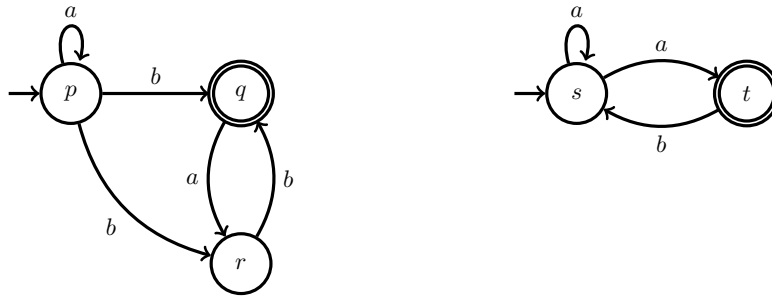
**Question 5 (2 + 2 + 2 = 6 points)**

Let  $L = \{w \in \{a, b\}^* \mid \text{between any two } b\text{'s of } w \text{ there is at least one } a\}$ .

- (a) Give an MSO formula  $\varphi$  such that  $L(\varphi) = L$ .
- (b) Give the minimal DFA for  $L$ .
- (c) Give a regular expression for  $L$ .

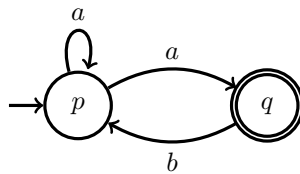
**Question 6 (5 points)**

Let  $A, B$  be the following Büchi automata over  $\Sigma = \{a, b\}$ . Construct a Büchi automaton  $C$  such that  $L_\omega(C) = L_\omega(A) \cap L_\omega(B)$  using the algorithm presented in the lecture notes.



**Question 7 (2 + 2 + 2 = 6 points)**

Consider the following Büchi automaton over  $\Sigma = \{a, b\}$ :



- (a) Sketch  $dag(aba^\omega)$  and  $dag(a(ab)^\omega)$ . You must build at least 5 levels.
- (b) Explain the conditions that an odd ranking must satisfy, and give an odd ranking of  $dag(aba^\omega)$ .
- (c) Using the definition of odd rankings, prove that  $dag(a(ab)^\omega)$  does not have any odd ranking.