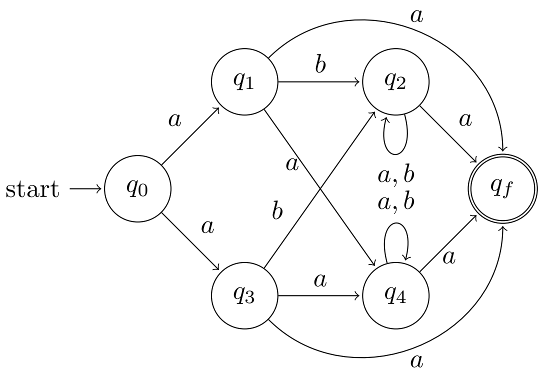
Technische Universität München **Winter term 2011/12**
I7
Prof. J. Esparza / J. Křetínský **27.10.2011**

Automata and Formal Languages – Homework 3

Due 3.11.2011.

Exercise 3.1

Consider the following NFA $\mathcal{A}$:



(a) Describe $\mathcal{L}(\mathcal{A})$.

(b) Determine the CSR of $\mathcal{A}$ using the algorithm presented in the lecture.

Exercise 3.2

Consider the partitioning algorithm from the lecture. Its while-loop clearly cannot be executed more than $|Q| - 1$ times. Show that this bound is tight, i.e. give an example where it is executed $|Q| - 1$ times. (Hint: It is sufficient to consider one-letter alphabet.)

Exercise 3.3

Let $L_i = \{w \in \{a\}^* \mid \text{ the length of } w \text{ is divisible by } i\ \}$.

(a) Construct an NFA for $L := L_4 \cup L_6$ with at most 11 states.

(b) Construct the minimal DFA for L.

Exercise 3.4

Let us consider $\Sigma = \{0, 1\}$ and the msbf encoding.

(a) Construct the minimal DFAs accepting the languages L_1, L_2, and L_3^2 defined below.

- $L_1 = \{w \mid \text{msbf}^{-1}w \mod 3 = 0\} \cap \Sigma^4$.
- $L_2 = \{w \mid \text{msbf}^{-1}w \text{ is a prime }\} \cap \Sigma^4$.
- $L_3^k = \{ww \mid w \in \Sigma^k\}$.

(b) How many states has the minimal DFA accepting L_3^k with respect to k?.

Exercise 3.5

For L_1, L_2 regular languages over an alphabet Σ, the *left quotient* of L_1 by L_2 is defined by

$$L_2 \backslash L_1 := \{v \in \Sigma^* \mid \exists u \in L_2 : \ uv \in L_1\}$$

(a) Use the fact that regular languages are closed under homomorphisms, inverse homomorphisms, concatenation and intersection to prove they are closed under quotienting.

(b) Given finite automata $\mathcal{A}_1, \mathcal{A}_2$, construct an automaton $\mathcal{A}$ such that

$$\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{A}_2) \backslash \mathcal{L}(\mathcal{A}_1)$$

(c) Is there any difference when taking the *right quotient* $L_1 / L_2 := \{u \in \Sigma^* \mid \exists v \in L_2 : \ uv \in L_1\}$?

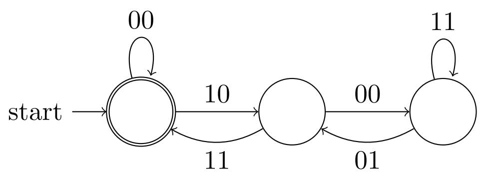
Exercise 3.6

Let L_1, L_2 be regular languages. Determine the inclusion relation between the following languages:

- L_1
- $(L_1 / L_2).L_2$
- $(L_1.L_2) / L_2$

Exercise 3.7

Consider the following FA $\mathcal{A}$ over the alphabet $\{00, 01, 10, 11\}$:



W.r.t. the msbf encoding, we may interpret any word $w \in \{00, 01, 10, 11\}^*$ as a pair of natural numbers $(X(w), Y(w)) \in \mathbb{N}_0 \times \mathbb{N}_0$. *Example*: (Underlined letters correspond to $Y(w)$.)

$$w = (00)^k 001011 \rightarrow (0\underline{0})^k 0\underline{0}1\underline{0}1\underline{1} \rightarrow (0^k011, 0^k001) \rightarrow (3, 1) = (X(w), Y(w))$$

(a) Show that $w \in \mathcal{L}(\mathcal{A})$ iff $X(w) = 3 \cdot Y(w)$.

(b) Construct the minimal DFA representing the language $\{w \in \{0,1\}^* \mid \text{msbf}^{-1}(w) \text{ is divisible by } 3\}$.

Exercise 3.8

Since regular languages are closed under complement and intersection, we may extend regular expressions with such primitives: If r and r' are regular expressions, then also $\overline{r}$ and $r \cap r'$ are regular expressions. A language $L \subseteq \Sigma^*$ is called *star-free*, iff there exists an extended regular expression r *without* a Kleene star such that $L = \mathcal{L}(r)$. For example, Σ^* is star-free, because it is the same as $\mathcal{L}(\overline{\emptyset})$. Show that $\mathcal{L}((01 + 10)^*)$ is star-free.