René Maseli
Prof. Dr. Roland Meyer

Theoretical Computer Science 1
Exercise 3

TU Braunschweig
Winter semester 2022/23

Release: 29.11.2022 Due: 09.12.2021, 09:45

Hand in your solutions to the Vips directory of the StudIP course until Friday, 09.12.2022 09:45 pm. You should provide your solutions either directly as .pdf file or as a readable scan/photo of your handwritten notes. Submit your results as a group of four.

Exercise 1: Properties of Regular Languages [8 points]

Show that the following statements are valid:

a) [4 points] The Kleene-Star is indeed a closure operator: $(L^*)^* = L^*$.

b) [4 points] For each pair of NFAs A and B, there are (other) NFAs $A \cup B$ und $A.B$, with $\mathcal{L}(A \cup B) = \mathcal{L}(A) \cup \mathcal{L}(B)$ and $\mathcal{L}(A.B) = \mathcal{L}(A).\mathcal{L}(B)$.

   Hint: Give a construction procedure for each NFA, that works for any A's and B's, and prove, that the respective languages are equal.

Exercise 2: Boolean Programs are regular [9 points]

Consider programs on boolean variables only. Expressions $e \in \text{Exp}$ are non-deterministicly evaluated on variable states $\sigma \in \{0,1\}^V$: For $v \in \{0,1\}$, $S_v(e)$ marks the set of variable states, on which e may return v. For a variable $x \in V$ is $S_v(x) = \{\sigma \in \{0,1\}^V \mid \sigma(x) = v\}$. I.e. $\sigma \in S_v(y \vee \neg z)$ holds, if and only if $v = \max(\sigma(y), 1 - \sigma(z)) \in \{0,1\}$. There is also the *havoc*-expression $*$, which non-deterministicly evaluates to both 0 and 1: $S_0(*) = S_1(*) = \{0,1\}^V$.

Let V be the set of variables in the boolean program. Some words of $\{s\} \cup (V \times \{0,1\})$ correspond to executions of the program. There is an NFA $\langle (B \times \{0,1\}^V), \langle b_0, 0^V\rangle, \rightarrow, \{f\} \times \{0,1\}^V\rangle$, whose language is exactly the set of words that correspond to halting executions. It starts on the initial block $i \in B$ with all variables set to 0 and accepts on the (sole) final block $f \in B$, regardless of the variables.

The transition $\langle b, \sigma\rangle \xrightarrow{\langle x,v\rangle} \langle c, \tau\rangle$ exists in the NFA, if and only if all $y \in V \setminus \{x\}$ satisfy $\sigma(y) = \tau(y)$, the value is $\tau(x) = v$, $b = [x := e]^\ell$ is an assignment, c is the successor of b and if $\sigma \in S_v(e)$.

The transition $\langle b, \sigma\rangle \xrightarrow{s} \langle c, \tau\rangle$ exists, if and only if $\sigma = \tau$, $b = [e]^\ell$ is the condition of a conditional or a loop, and either

- c is the first else-Block or if no such exists, the successor of b, and $\sigma \in S_0(e)$.
- c is the first then-Block or the first inner loop block and $\sigma \in S_1(e)$.

a) [8 points] Consider the following program P with blocks $B = \{b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7\}$ and purely-boolean variables $V = \{x, y, z\}$.

$[x := *]^0$
while $[\neg x \wedge \neg z]^1$ **do**
  $[y := \neg x \vee \neg z]^2$
  $[x := *]^3$
  if $[x \wedge y]^4$ **then**
    $[y := y \vee \neg z]^5$
  end if
  $[z := \neg z \vee \neg y]^6$
end while
$[\text{skip}]^7$

Construct the finite automaton A_P that accepts words of $\{s\} \cup V \times \{0, 1\}$: $\Sigma = \{s, x0, x1, y0, y1, z0, z1\}$.

$\varepsilon \notin \mathcal{L}(A_P)$, since every execution must pass through Block b_0.

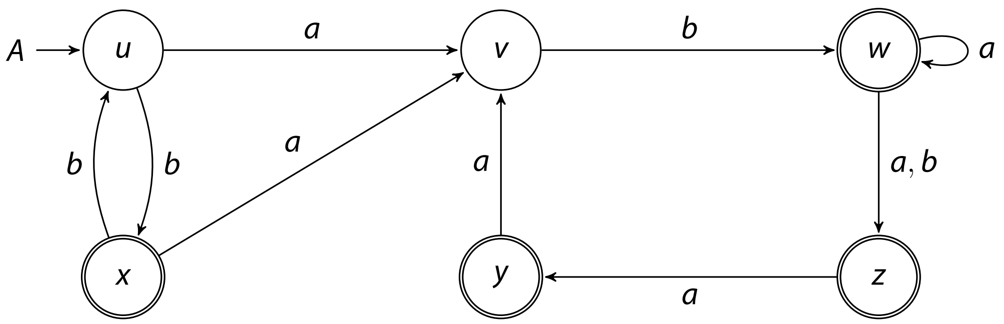
$x1.s \notin \mathcal{L}(A_P)$, since executions always start with $z = 0$ and therfore have to iterate at least once.

$x1.s.y0.x1.s.z1.s \in \mathcal{L}(A_P)$, because there is an execution that first reads 1 and then 0, breaking the loop.

b) [1 Punkt] Is your automaton A_P *partially* deterministic (missing transitions just have to lead into a new state $\varnothing$)?

Exercise 3: NFA to REG using Arden's Rule [10 points]

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



a) [1 point] Formulate the equation system associated with A.

b) [3 points] Find a regular expression for $\mathcal{L}(A)$ by solving the equation system using Arden's Rule.

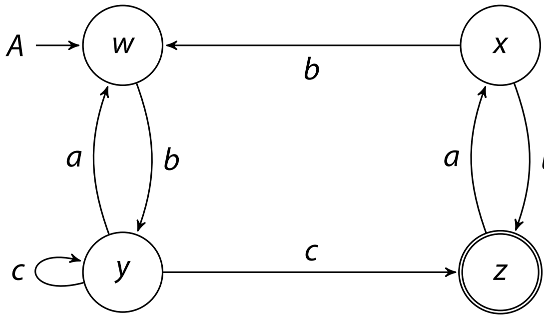
c) [2 points] Give expressions for all other variables of the equation system.

d) [3 points] Find a DFA B with $\mathcal{L}(B) = \mathcal{L}(A)$ using the construction by Rabin & Scott.

e) [1 point] Give a regular expression for $\overline{\mathcal{L}A}$.

Exercise 4: Homomorphisms [8 points]

Examine the following NFA A over the alphabet $\Sigma = \{a, b, c\}$:



Consider the homomorphism $f: \Sigma \to \{0, 1\}$.

$$f(a) = \varepsilon$$
$$f(b) = 10$$
$$f(c) = 01$$

a) [3 points] Construct the image-automaton $f(A)$ with $\mathcal{L}(f(A)) = f(\mathcal{L}(A))$.

b) [1 point] Show $1001011010100110 \in \mathcal{L}(f(A))$ by giving a corresponding run through A.

Consider the homomorphism $g: \{d, e\} \to \Sigma$.

$$g(d) = bcc$$
$$g(e) = ab$$

c) [3 points] Construct the co-image-automaton $g^{-1}(A)$ with $\mathcal{L}(g^{-1}(A)) = g^{-1}(\mathcal{L}(A))$.

c) [1 point] Show *deeded* $\in \mathcal{L}(g^{-1}(A))$ by giving a corresponding run through A.