Prof. Dr. Roland Meyer	**Theoretical Computer Science** **Exercise Sheet 3**	TU Braunschweig
René Maseli		Winter Semester 2025/26

Release: 2025-11-24 Due: 2025-12-04 23:59

Homework Exercise 3.1: Regular languages and finite automata [5 points]
Let Σ be an alphabet and FA(Σ) be the class of finite automata over Σ. Show that the following statements are valid:

a) [1 point] $\forall A, B \in \text{FA}(\Sigma) : \exists A.B \in \text{FA}(\Sigma) : \mathcal{L}(A.B) = \mathcal{L}(A).\mathcal{L}(B)$

Hint: Give construction procedures, that work for any *A*'s and *B*'s, and show, that the respective languages are equal.

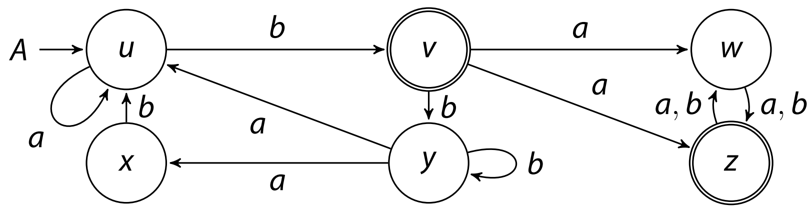
b) [2 points] $\forall A \in \text{FA}(\Sigma) : \exists A^{\text{re}} \in \text{FA}(\Sigma) : \mathcal{L}(A^{\text{re}}) = \mathcal{L}(A)^{\text{re}} := \{\, a_n \dots a_1 \mid a_1 \dots a_n \in \mathcal{L}(A) \,\}$

c) [2 points] Show that the Kleene-Star is indeed a closure operator: $(L^*)^* = L^*$.

Hint: Prove and use the following lemmata:

a) $\forall i \in \mathbb{N} : L^*.L^i \subseteq L^*$ b) $L^*.L^* \subseteq L^*$ c) $\forall i \in \mathbb{N} : (L^*)^i \subseteq L^*$.

Homework Exercise 3.2: Automaton to Regular Expression [4 points]
Consider the following NFA *A* over the alphabet $\{a, b\}$.



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

b) [2 points] Find a regular expression for $\mathcal{L}(A)$ by solving the equation system using Arden's Rule. Give expressions for all other variables of the equation system.

c) [1 point] Describe what happens in this procedure, if no accepting state is reachable from the initial state. How does this affect the solution space of the equation system?

Homework Exercise 3.3: Rabin & Scott [3 points]

Let $A = \langle Q, q_0, \rightarrow, Q_F \rangle$ be an NFA over Σ, and $\mathcal{P}(A) = \langle \mathcal{P}(Q), Q_0, \rightarrow_{\mathcal{P}(A)}, Q'_F \rangle$ be the automaton constructed via the Rabin-Scott powerset construction, with $Q_0 = \{q_0\}$, $X \xrightarrow{a}_{\mathcal{P}(A)} \{ q \in Q \mid \exists p \in X \colon p \xrightarrow{a} q \}$ for all $X \subseteq Q$ and $a \in \Sigma$, and $Q'_F = \{ X \subseteq Q \mid X \cap Q_F \neq \emptyset \}$.

The task of this exercise is to proof Theorem 3.18. Towards this, proceed as follows:

a) [1 point] Show by induction on i: For every run $q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \ldots \xrightarrow{a_i} q_i$ of A the (unique) run $Q_0 \xrightarrow{a_1}{}' Q_1 \xrightarrow{a_2}{}' \ldots \xrightarrow{a_i}{}' Q_i$ of $\mathcal{P}(A)$, which reads the same word, satisfies $q_i \in Q_i$.

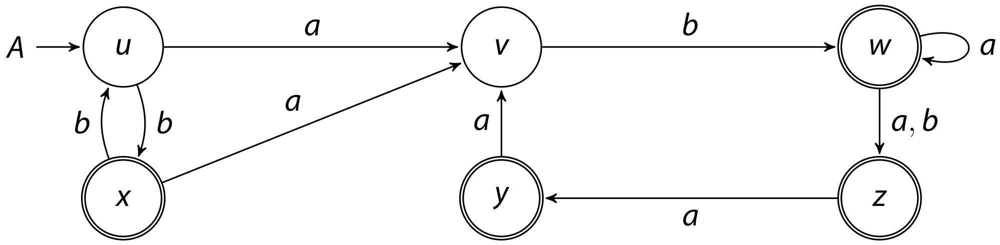
b) [1 point] Show by induction on i: For every run $Q_0 \xrightarrow{a_1}{}' Q_1 \xrightarrow{a_2}{}' \cdots \xrightarrow{a_i}{}' Q_i$ of $\mathcal{P}(A)$ and every state $q_i \in Q_i$ there exists a run $q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_i} q_i$ of A, reading the same word and stops in q_i.

c) [1 point] Using the partial results of a) and b), prove that $\mathcal{L}(A) = \mathcal{L}(\mathcal{P}(A))$ holds.

Homework Exercise 3.4: Powerset Construction [5 points]

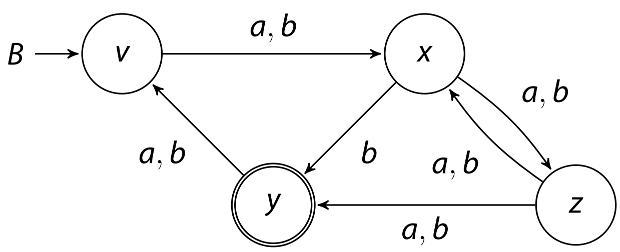
Betrachten Sie die folgenden NFAs A und B über $\Sigma = \{a, b\}$.

a) [3 points] Construct an automaton $\overline{A_{\text{det}}}$ with $\mathcal{L}(\overline{A_{\text{det}}}) = \overline{\mathcal{L}(A)}$. Therefore, determinize A, that is, find a DFA A_{det} with $\mathcal{L}(A_{\text{det}}) = \mathcal{L}(A)$ by using the Rabin-Scott powerset construction.



Hint: You can restrict to the states reachable from the initial state $\{q_0\}$. For this, start with $\{q_0\}$ as the only state and then iteratively construct for the current set of states all possible direct successors, until no more states are added.

b) [2 points] Consider the words $w_1 = babab$, $w_2 = abbbaa$ and $w_3 = bbbbaaa$. Determine, whether or not $w_1, w_2, w_3 \in \mathcal{L}(B)$ is valid, by giving the respective runs of the Powerset Automaton.



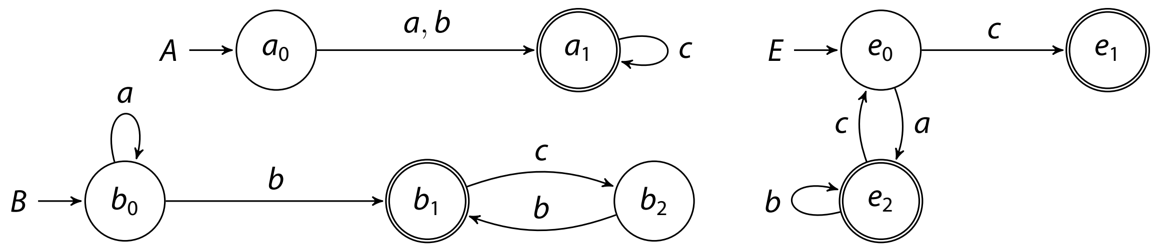
Tutorial Exercise 3.5:

Goal is a finite automaton for the following regular expression

$$(c + a(b + ca)^*(\varepsilon + cc))((a + b)c^* + a^*b(cb)^*)^* \text{ over } \Sigma = \{a, b, c\} .$$

Remark: If suffices to draw the state graph. In this lecture, the operations have the following precedence: (*), (.), ($\cup$)

Consider the automata A for $(a + b)c^*$ and B for $a^*b(cb)^*$. Give an automaton C for $\mathcal{L}(A) \cup \mathcal{L}(B)$, as well as an automaton D for $\mathcal{L}(C)^*$.



Consider further the automaton E for $c + a(b + ca)^*(\varepsilon + cc)$ and give an automaton F for $\mathcal{L}(E).\mathcal{L}(D)$

.

Tutorial Exercise 3.6:

Check whether the following problems can be considered as problems over regular languages. Explain your answer by giving a regular expression or a finite automaton, if possible, or by a brief argument, that the language is indeed not regular. Correctness proofs are not needed.

Assume the alphabet $\Sigma = L \cup U \cup D \cup S \cup W$, partitioned into lower-case letters L, upper-case letters U, digits D, special characters S and white spaces W.

a) Does the input have at least 3 symbols and at most 18?

b) Does each class of non-space symbols (L, U, D and S) occur at least once?

c) *Parenthesization:* Is the input text correctly parenthesized, i.e. does every opening parenthesis have a matching closing parenthesis and vice versa? `(ri)(gh)t`, `R(i(g)h)t` are correct, but `w(r)on)g` and `W(r)o(n(g` are not.

d) *String literals:* We may enclose sequences with $'$ $\in S$ and *escape* each directly following symbol with \ $\in S$. Does every non-escaped opening `'` have a matching closing `'` and are all white space and special character either enclosed or escaped? (e.g. `'no'issue'with'\'\\'` or `'Robert\');DROP TABLE Students;--'`)

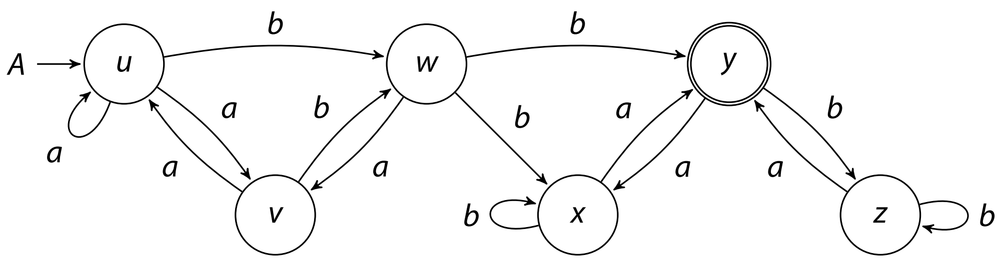
e) *Tables:* Do all rows (separated by newlines \n $\in W$) have the same number of cells (separated by commata `,` $\in S$)?

f) *Comments:* Are all comment sections closed?

Those start with either $// \in S^2$ or with $/* \in S^2$. The formers end with the newline \n $\in W$; the latters end with $*/ \in S^2$. Nested comments do not work, therefore ///*\n, /*/**/ and /*//*/ are elements of the language.

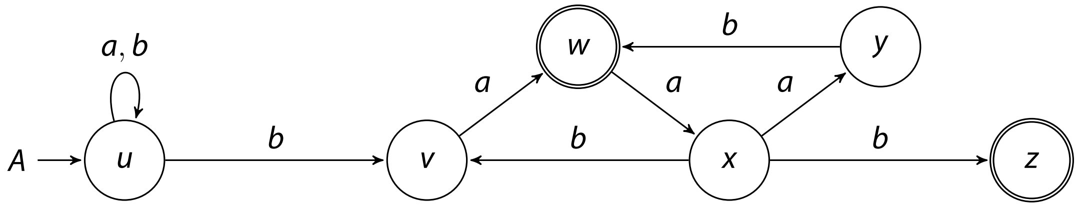
Tutorial Exercise 3.7:

Consider the following finite automaton. Compute an equivalent regular expression with the use of Arden's Rule.



Tutorial Exercise 3.8:

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



Formulate the equation system associated with A. Find a regular expression for $\mathcal{L}(A)$ by solving the system using Arden's Rule.

Tutorial Exercise 3.9:

Consider the Automaton from exercise 2. Construct the equivalent deterministic automaton by the method of Rabin & Scott.

Tutorial Exercise 3.10:

Apply the construction of Rabin & Scott's, to get an equivalent, deterministic finite automaton A_{det} for the following automaton A: Only draw the reachable states of A_{det}.