Übungsblatt 2

Aufgabe 1 Sei Σ ein (endliches) Alphabet.

- Geben Sie zunächst die Funktion leaves : $\mathcal{E}_\Sigma \to \mathbb{N}$ induktiv an, welche die Terminalzeichen eines regulären Ausdrucks zählt.
- Mit $\Sigma_n = \mathbb{N} \times \Sigma$ bezeichnen wir das (unendliche) Alphabet, das aus durchnummerierten Terminalzeichen besteht. Definieren Sie das Durchnummerieren num : $\mathcal{E}_\Sigma \to \mathcal{E}_{\Sigma_n}$ eines regulären Ausdrucks. Verwenden Sie hierzu eine Hilfsfunktion num$'$: $\mathbb{N} \to \mathcal{E}_\Sigma \to \mathcal{E}_{\Sigma_n}$, welche die Startnummerierung als Parameter erhält.

Aufgabe 2 Seien $e_1, e_2 \in \mathcal{E}_{\{a,b,c\}}$ gegeben durch

- $e_1 = a^*|(bc)$
- $e_2 = a^*(b|c)^*$

Wandeln Sie zunächst die regulären Ausdrücke in ihre durchnummerierten Formen um, indem Sie die vorher definierte Funktion num verwenden. Bestimmen Sie anschließend empty, first, last und next aus dem Berry-Sethi-Verfahren.

Aufgabe 3 Sei Σ ein endliches Alphabet und $r \in \mathcal{E}_\Sigma$. Definieren Sie die Funktionen empty, first, last und next aus dem Berry-Sethi-Verfahren für r^+. Unterscheidet sich eine Funktion von r^*r?

Aufgabe 4 Sei $e_n = (a|b)^*a((a|b)\{n,n\}) \in \mathcal{E}_{\{a,b\}}$ für $n \in \mathbb{N}$.

- Bestimmen Sie $[\![e_n]\!]$.
- Konstruieren Sie NDEAs A_n mit $L(A_n) = [\![e_n]\!]$.
- Zeigen Sie, dass jeder DEA B_n mit $L(B_n) = L(A_n)$ mindestens 2^n Zustände besitzen muss.

  Hinweis: Betrachten Sie alle paarweise verschiedenen Wörter aus $\{a, b\}^n$ und verwenden Sie das Schubfachprinzip, um zu zeigen, dass der Automat jedes dieser Wörter unterscheiden muss.