Universität Siegen
Lehrstuhl Theoretische Informatik
Markus Lohrey

Compilerbau I
SS 2022

Übungsblatt 5

Aufgabe 1 Geben Sie reguläre Ausdrücke an für die gültigen...

- Kalenderdaten in 2022
- IPv4-Adressen
- Hostnamen

Recherchieren Sie ggf. nach ihren Definitionen in RFCs.

Lösung:

1) $e_1 = [1-9], e_2 = [0-9], e_3 = [0-8]$

$$\begin{aligned} e_4 =& 0e_1 \mid 1e_2 \mid 2e_2 \mid 3(0 \mid 1) \\ e_5 =& 0e_1 \mid 1e_2 \mid 2e_2 \mid 30 \\ e_6 =& 0e_1 \mid 1e_2 \mid 2e_3 \\ r_k =& e_4.(01 \mid 03 \mid 05 \mid 07 \mid 08 \mid 10 \mid 12).2022 \mid \\ & e_5.(04 \mid 06 \mid 09 \mid 11).2022 \mid \\ & e_6.02.2022 \end{aligned}$$

über dem Alphabet $\Sigma = \{., 0, \ldots, 9\}$.

2) 32-Bit-Form: $r_i = ((0 \mid 1)^8.)^3(0 \mid 1)^8$ oder umgewandelt in 10er-System-Form (beachte: $1111\,1111_2 = 255_{10}$):

$$\begin{aligned} e_7 &= e_2 \mid e_1e_2 \mid 1e_2e_2 \mid 2[0-4]e_2 \mid 25[0-5] \\ r_i' &= (e_7.)^3e_7 \end{aligned}$$

über dem Alphabet $\Sigma = \{., 0, 1\}$ bzw. $\Sigma = \{., 0, \ldots, 9\}$. Ohne Subnetze oder dergleichen (nur die IP).

3) Was sind Hostnamen? Wir nehmen eine leicht vereinfachte Definition. Das Alphabet ist $\Sigma = \{., 0, \ldots, 9, a, \ldots, z, -\}$. Sei

$$\begin{aligned} e_8 &= [a-z] \mid [0-9] \\ e_9 &= e_8(e_8 \mid -)\{0, 61\}e_8 \mid e_8 \\ r_h &= e_9.e_9(.e_9)^?(.e_9)^? \end{aligned}$$

Es gelten folgende Regeln:

- Es gibt 2-4 Blöcke, bestehend aus Zeichen aus Σ ohne Punkt. Die Blöcke werden durch Punkte getrennt.
- Jeder Block besteht aus 1-63 Zeichen.
- Erstes und letztes Zeichen in einem Block darf kein $-$ sein.
- Insgesamt darf ein Hostname höchstens 255 Zeichen besitzen ($63 \cdot 4 + 3 = 255$).
- Hostnamen haben also die Form en.wikipedia.org
- Wir ignorieren Sonderregeln wie Großschreibung usw.

Aufgabe 2 Das Berry-Sethi-Verfahren konstruiert einen NFA, der einen einzigen Startzustand besitzt. Wandeln Sie das Verfahren so um, dass der NFA stattdessen einen einzigen Endzustand besitzt. Verwenden Sie hierzu die bereits bekannten Funktionen empty, first, last und next.

Lösung:
Sei $r \in \mathcal{E}_\Sigma$ und $e = \text{num}(r)$ die durchnummerierte Version von r. Wir definieren den Automat zu e wie folgt:

$$A = (Q, \Sigma, \Delta, S, F)$$

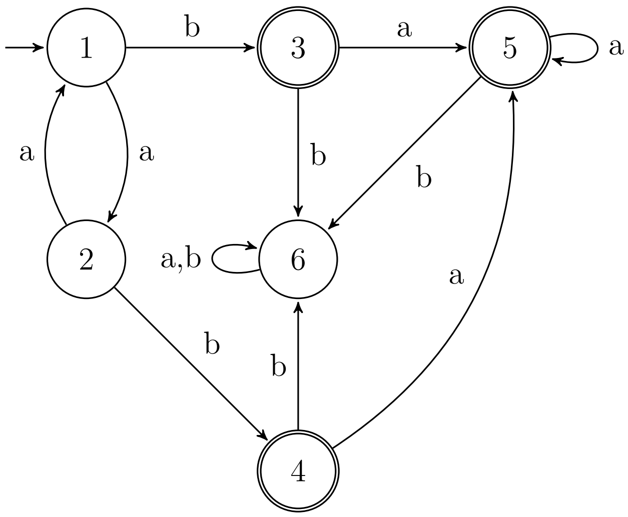
$$\begin{aligned}
Q &= \{1, \ldots, \ell(e)\} \cup \{\#\} \\
S &= \{i \mid \exists a.[i, a] \in \text{first}(e)\} \cup \{\# \mid \text{empty}(e) = t\} \\
F &= \{\#\} \\
\Delta &= \{(i, a, \#) \mid [i, a] \in \text{last}(e)\} \\
&\quad \cup \{(i, a, j) \mid \exists b.[j, b] \in \text{next}_e([i, a])\}
\end{aligned}$$

Aufgabe 3 Wiederholen Sie den DFA-Minimierungsalgorithmus aus dem GTI-Skript (Folie 165).

(a) Gegeben sei ein DFA $A = (Q, \Sigma, \delta, \{q_0\}, F)$ und eine Klassifizierung $r\colon Q \to \mathbb{N}$. Beschreiben Sie, wie Sie den Algorithmus aus der GTI-Vorlesung erweitern müssen, damit $\equiv_r$ berechnet wird.

Lösung:
Punkt 2 muss folgendermaßen geändert werden: Markiere alle Paare $\{z, z'\}$ mit $r(z) \neq r(z')$. Begründung: Damit trennt man initial auch die Zustände von F in die einzelnen F_i. In allen folgenden Schritten wird berücksichtigt, dass Zustände nicht erkennungsäquivalent sind, wenn man durch Einlesen eines Wortes w in verschiedene F_i kommt.

(b) Minimieren Sie folgenden DFA $(Q, \Sigma, \delta, I, F)$ gegeben durch



mit der Klassifizierung $r\colon Q \to \mathbb{N}$, wobei

$$\begin{aligned} &r(3) = r(4) = 1, \\ &r(5) = 2, \\ &r(q) = 0 \text{ für } q \in Q \setminus F. \end{aligned}$$

Lösung:
Wir erhalten folgende Tabelle:

2		-	-	-	-
3	1	1	-	-	-
4	1	1		-	-
5	1	1	1	1	-
6	2	3	1	1	1
	1	2	3	4	5

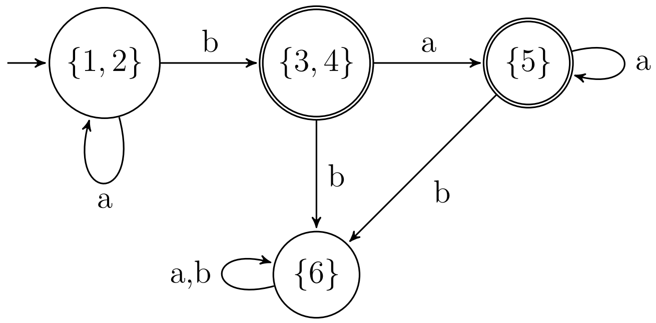
Die Schritte im Einzelnen sind:

(1) Markiere alle Zustandspaare $\{z, z'\}$ mit $r(z) \neq r(z')$.

(2) Markiere $\{1, 6\}$, weil $\delta(1, b) = 3$, $\delta(6, b) = 6$ und $\{3, 6\}$ markiert.

(3) Markiere $\{2, 6\}$, weil $\delta(2, b) = 4$, $\delta(6, b) = 6$ und $\{4, 6\}$ markiert.

Es sind keine weiteren Schritte möglich. Damit sind 1 und 2, und 3 und 4 äquivalent und es ergibt sich der folgende Automat:



mit der Klassifizierung $r'\colon Q \to \mathbb{N}$, wobei

$$\begin{aligned} r'(\{3,4\}) &= 1, \\ r'(\{5\}) &= 2, \\ r'(\{1,2\}) &= 0, \\ r'(\{6\}) &= 0. \end{aligned}$$

Aufgabe 4 Gegeben sei folgende Übergangsfunktion eines DFAs:

	0	1	2	3	4	5
a	1	4	5	4	4	3
b	2	3	4	0	4	2

(a) Wenden Sie das Displacement-Verfahren an, um eine Übergangstabelle mit nur einer Zeile zu erhalten. Geben Sie die displacement-Funktion sowie die resultierende Tabelle inklusive der valid-Zeile an. Was ist die kleinste Anzahl an Spalten, die Sie erreichen können?

Lösung:
Wir wählen Default = 4 und erhalten

	0	1	2	3	4	5
a	1		5			3
b	2	3		0		2

Dann verschieben wir die Zeile für a um 2 nach rechts und erhalten

	0	1	2	3	4	5	6	7
a			1		5			3
b	2	3		0		2		

Damit ergibt sich displacement$(a) = 2$, displacement$(b) = 0$ und

	0	1	2	3	4	5	6	7
A	2	3	1	0	5	2		3
valid	b	b	a	b	a	b		a

Wir erhalten 8 Spalten durch das Verschieben der Zeile für a. Die Zeile für b hingegen müsste man um mindestens 3 nach rechts verschieben, damit beide Zeilen übereinander passen, also würde man mindestens 9 Spalten erhalten.

(b) Sei 0 der Startzustand des Automaten. Geben Sie die Konfigurationsfolge für die Eingabe $ababa$ an, indem Sie die Übergangsfunktion von Folie 78 verwenden.

Lösung:
Eine Konfigruation notieren wir als Paar bestehend aus dem aktuellen Zustand und dem Restwort.

Die Startkonfiguration ist $(0, ababa)$.

- Wegen displacement$(a) = 2$ schauen wir in Spalte $0 + 2 = 2$. Da valid$(2) = a$ und $A(2) = 1$, ist die nächste Konfiguration $(1, baba)$.
- Wegen displacement$(b) = 0$ schauen wir in Spalte $1 + 0 = 1$. Da valid$(1) = b$ und $A(1) = 3$, ist die nächste Konfiguration $(3, aba)$.
- Wegen displacement$(a) = 2$ schauen wir in Spalte $3 + 2 = 5$. Da valid$(5) = b$, müssen wir nach Default gehen. Also ist die nächste Konfiguration $(4, ba)$.
- Wegen displacement$(b) = 0$ schauen wir in Spalte $4 + 0 = 4$. Da valid$(4) = a$, müssen wir nach Default gehen. Also ist die nächste Konfiguration $(4, a)$.
- Wegen displacement$(a) = 2$ schauen wir in Spalte $4 + 2 = 6$. Da valid(6) leer ist, müssen wir nach Default gehen. Also ist die nächste Konfiguration $(4, \varepsilon)$.