Lehrstuhl für Softwaretechnik und Programmiersprachen
Professor Dr. Michael Leuschel

# Compilerbau - Wintersemester 2021/22

## Übungsblatt 1 - Musterlösung

### Aufgabe 1.1

1. Sei $\gamma := (0|1)^*a|1b$. Dann ist
$$L(\gamma) = \{ua \mid u \in \{0,1\}^*\} \cup \{1b\}\,.$$

2. Sei $\gamma := 0((\epsilon|0)1^*)^*0$. Dann ist
$$\gamma = 0(0|1)^*0.$$
Es folgt
$$L(\gamma) = \{0u0 \mid u \in \{0,1\}^*\}\,.$$

3. Sei $\gamma := 010^*10^*10^*(10)^*$. Dann ist
$$L(\gamma) = \{01u1v1wx \mid u, v, w \in \{0\}^*, x \in \{10\}^*\}\,.$$

### Aufgabe 1.2

1. Seien $\gamma_1 := ((\epsilon|0)1^*)^*$ und $\gamma_2 := (01)^*$. Offensichtlich ist $0 \in L(\gamma_1)$, aber $0 \notin L(\gamma_2)$. Folglich ist $L(\gamma_1) \neq L(\gamma_2)$.

2. Seien $\gamma_1 := (0^*|1|1^*)$ und $\gamma_2 := (0^*1^*)$. Es ist $01 \in L(\gamma_2)$, aber $01 \notin L(\gamma_1)$. Damit ist $L(\gamma_1) \neq L(\gamma_2)$.

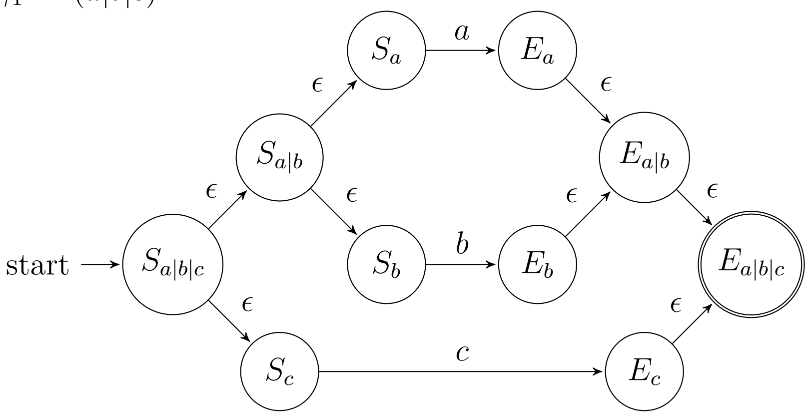
3. Seien $\gamma_1 := (0|1|1^*)^*$ und $\gamma_2 := (0^*1^*)^*$. Beide Ausdrücke lassen sich umformen:
$$\gamma_1 = (0|1^*)^* = (0|1)^*$$
$$\gamma_2 = (0|1)^*$$
Somit gilt $L(\gamma_1) = L(\gamma_2)$.

4. Seien $\gamma_1 := (0^*|1^*)$ und $\gamma_2 := (1^*|0^*)$. Offensichtlich ist $L(\gamma_1) = L(\gamma_2)$, da $|$ kommutativ ist.

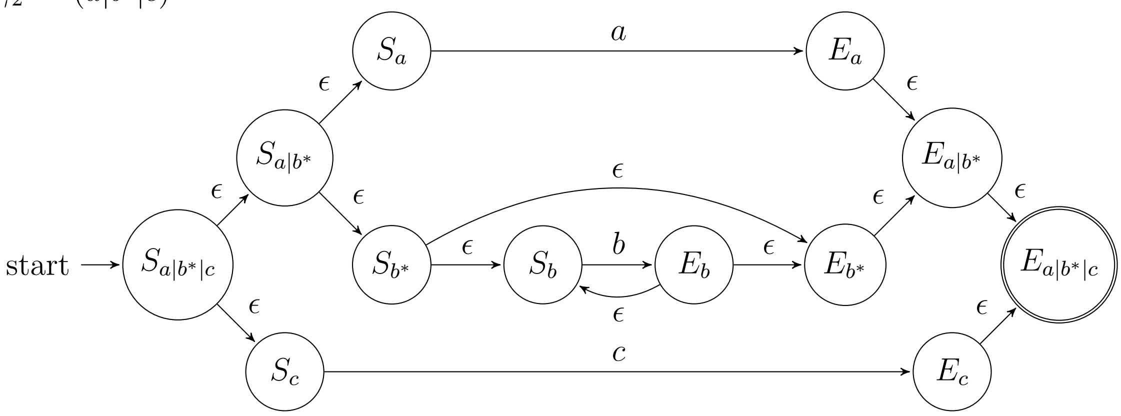
5. Seien $\gamma_1 := ((\epsilon|0)^*|0|1^*)^*$ und $\gamma_2 := (\epsilon|0|1^*)^*$. Beide Ausdrücke lassen sich vereinfachen:
$$\gamma_1 = (0^*|0|1^*)^* = (0^*|1^*)^* = (0|1)^*$$
$$\gamma_2 = (0|1^*)^* = (0|1)^*$$
Somit gilt $L(\gamma_1) = L(\gamma_2)$.

**Aufgabe 1.3**
Die NFAs werden, wie in der Aufgabenstellung verlangt, nach dem Drachenbuch konstruiert, obwohl es einfachere intuitivere Lösungen gibt. Es gilt Linksassoziativität bei gleichen Symbolen.

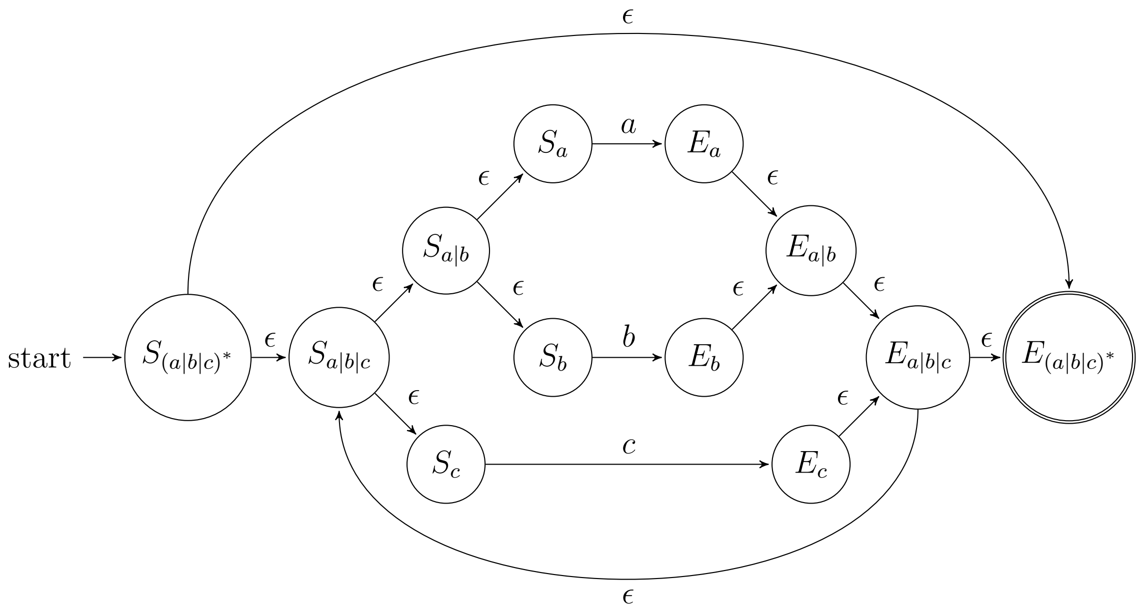
1. $\gamma_1 \coloneqq (a|b|c)$



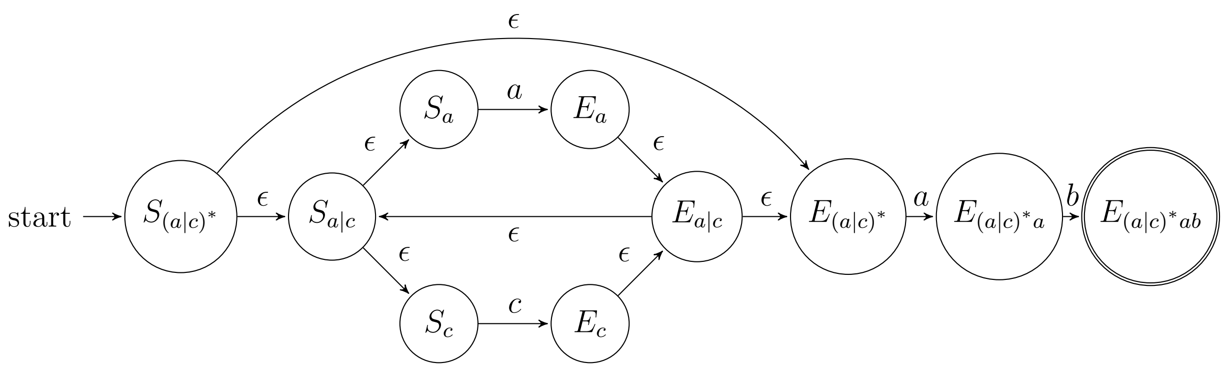

2. $\gamma_2 \coloneqq (a|b^*|c)$




3. $\gamma_3 \coloneqq (a|b|c)^*$



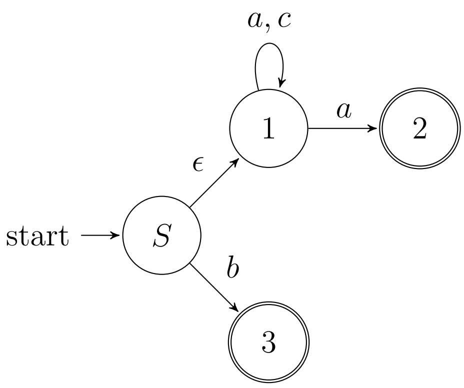

4. $\gamma_4 \coloneqq (a|c)^*ab$



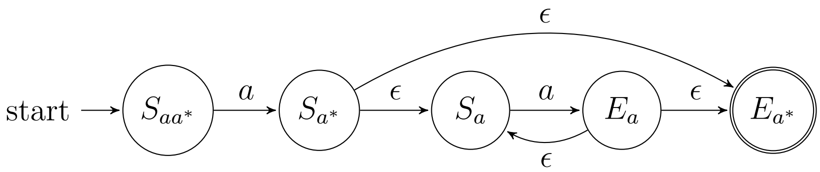

5. $\gamma_5 \coloneqq (a^+|c)^*a|b$
   Da im Drachenbuch für $a^+$ kein kanonischer Automat aufgelistet ist, wird der intuitive NFA vorgezogen. Dazu wird $\gamma_5$ erst vereinfacht:

$$\gamma_5 = (a|c)^*a|b$$

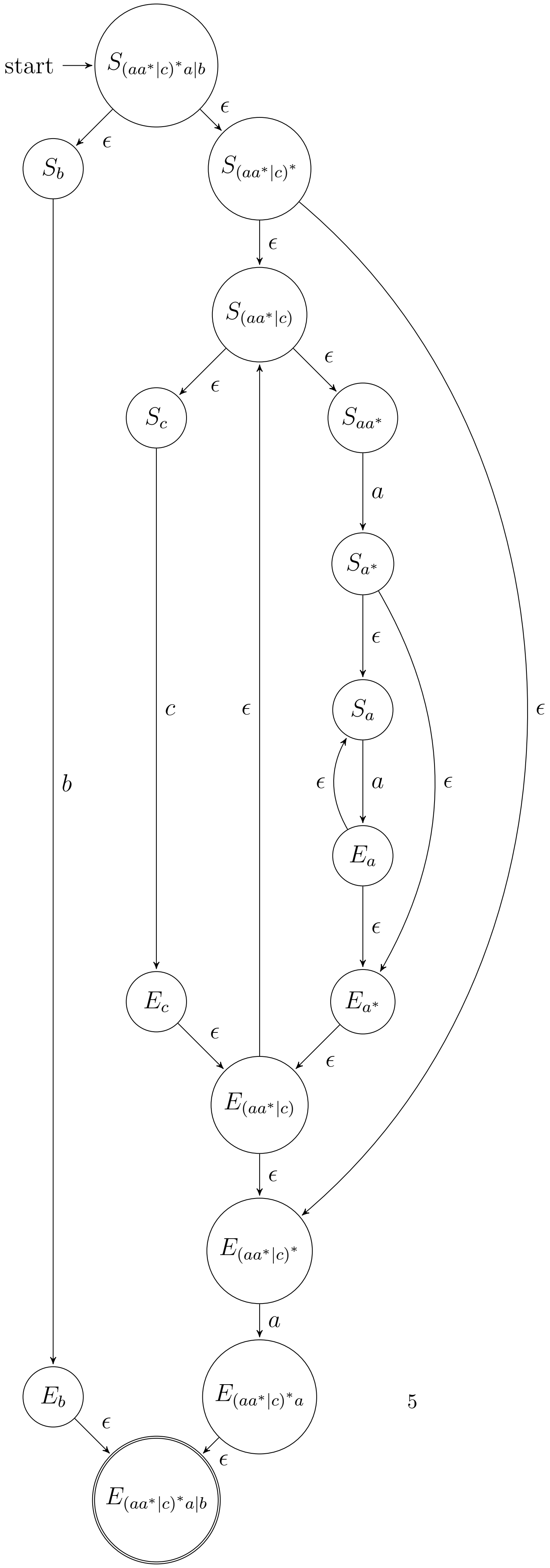
   Somit sieht der intuitive Automat folgendermaßen aus:




Der kanonische NFA für $a^+ = aa^*$ sieht folgendermaßen aus:

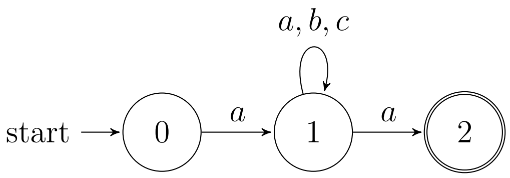



Mit diesem können wir nun einen NFA für $\gamma_5 = (aa^*|c)^*a|b$ bilden:

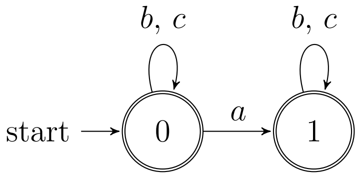

start
$S_{(aa^*|c)^*a|b}$
$S_b$
$S_{(aa^*|c)^*}$
$S_{(aa^*|c)}$
$S_c$
$S_{aa^*}$
$S_{a^*}$
$S_a$
$E_a$
$E_c$
$E_{a^*}$
$E_{(aa^*|c)}$
$E_{(aa^*|c)^*}$
$E_b$
$E_{(aa^*|c)^*a}$
$E_{(aa^*|c)^*a|b}$
$\epsilon$
$a$
$b$
$c$

**Aufgabe 1.4**
Die Aufgabenstellung wird so interpretiert, dass man einen regulären Ausdruck und einen endlichen Automaten finden muss, der alle möglichen Wörter, die die Eigenschaft erfüllen, erzeugen kann. Bei dieser Lösung der Aufgabe wird die intuitive Konstruktion der Automaten vorgezogen.

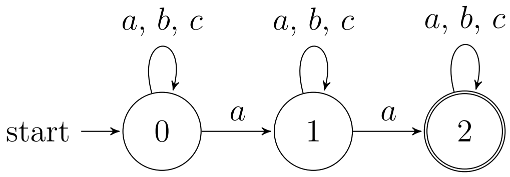
1. Sei $\gamma \coloneqq a(a|b|c)^*a$. Dann ist $N(\gamma) =$




2. Sei $\gamma \coloneqq (b|c)^*(a|\epsilon)(b|c)^*$. Dann ist $N(\gamma) =$

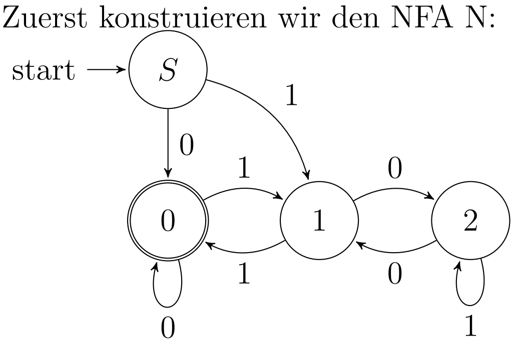



3. Sei $\gamma \coloneqq (a|b|c)^*a(a|b|c)^*a(a|b|c)^*$. Dann ist $N(\gamma) =$



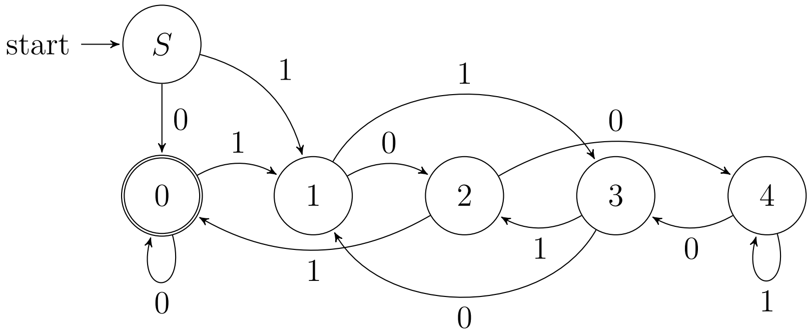

4. Mit regulären Ausdrücken nicht möglich.

5. Zuerst konstruieren wir den NFA N:




   Der reguläre Ausdruck ist
   $$\gamma \coloneqq \alpha\alpha^*$$
   mit
   $$\alpha \coloneqq 0|1(01^*0)^*1.$$

6. Zuerst konstruieren wir den NFA N:




Der reguläre Ausdruck ist

$$\gamma \coloneqq (0|1\alpha)\,(0|1\alpha)^*$$

mit

$$\alpha \coloneqq (\beta 0)^*\,(\beta 11)\,,$$

wobei

$$\beta \coloneqq (001^*0|1)\,(101^*0)^*.$$