# Prüfung zur Vorlesung „Compilerbau“

## WS 2021/22 / 14. Februar 2022

**Vorname:** ______________________________

**Nachname:** ______________________________

**Matrikelnummer:** ______________________________

| Aufgabe | Punktzahl | Erreicht |
|---|---|---|
| 1 | 10 | |
| 2 | 12 | |
| 3 | 12 | |
| 4 | 6 | |
| 5 | 10 | |
| 6 | 0 | |
| $\Sigma$ | 50 | |

## Generelle Hinweise:

- Prüfungsdauer: **60 Minuten**. Die Prüfung findet als Take-Home-Exam von **9 bis 10 Uhr** statt.
- Die Lösungen müssen mit der Hand geschrieben werden. Schreiben Sie bitte deutlich. Unleserliche Lösungen sind ungültig.
- Es ist nicht notwendig, die Klausur auszudrucken: Sie können Ihre Lösungen gerne auf eigene (einfarbig weiße, linierte oder karierte) DIN-A4-Blätter schreiben.
- Notieren Sie bitte **auf jedem Blatt**, das Sie verwenden, Ihren **Namen**, Ihre **Matrikelnummer** und die **Aufgabe**, die Sie bearbeiten.
- Die fertigen Lösungen **scannen oder fotografieren** Sie. Achten Sie auf gute Lesbarkeit. Wir empfehlen die kostenlose App Adobe Scan zum Einscannen der Seiten. Senden Sie bitte Ihre Lösungen im PDF-Format.
- Die Benutzung eines Tablets zur Anfertigung Ihrer handgeschriebenen Lösungen ist leider nicht erlaubt!
- Der Name der PDF muss die folgende Form haben:
  Nachname_Vorname_Matrikelnummer.pdf oder
  Nachname Vorname Matrikelnummer.pdf
- Ihre Lösungen müssen bis **spätestens 10:20 Uhr** am 14. Februar 2022 (heute) bei der folgenden Adresse ankommen: michael.figelius@uni-siegen.de
- Zusammen mit Ihren Lösungen schicken Sie eine ausgefüllte und unterschriebene Erklärung über die eigenständige Erbringung der Prüfungsleistung.
- Alle Hilfsmittel sind erlaubt bis auf die Hilfestellung durch eine andere Person.
- Bei absoluten Notfällen können Sie uns unter folgender Telefonnummer erreichen:
  [redacted]
  Wir können nicht garantieren, dass diese Nummer auch stets erreichbar ist.

Zur Erinnerung:

**Definition.** *Sei $k \in \mathbb{N}$. Eine Grammatik $G = (N, \Sigma, P, S)$ heißt* LL($k$)-Grammatik, *wenn für jede Linkssatzform $S \to^* wA\beta$, wobei $w \in \Sigma^*$, $A \in N$ und $\beta \in (\Sigma \cup N)^*$ gilt: Für jedes Paar von Produktionen $A \to \alpha_1, A \to \alpha_2 \in P$ mit $\alpha_1 \neq \alpha_2$ gilt, dass* $\mathrm{First}_k(\alpha_1\beta) \cap \mathrm{First}_k(\alpha_2\beta) = \emptyset$.

**Aufgabe 1.** (10 Punkte) Beantworten Sie die folgenden Fragen. Jede Teilaufgabe bringt Ihnen bei vollständiger und korrekter Beantwortung zwei Punkte.

(1) Was ist die Aufgabe eines Compilers? Welchen Input bzw. Output hat ein Parser?

> **Lösung:** Der Compiler generiert aus einem Programmtext einen Code (Maschinenprogramm). Das Maschinenprogramm wiederum kann auf eine Eingabe (effizient) angewendet werden.
>
> Input Parser: Tokenstrom
>
> Output Parser: Syntaxbaum

(2) Sei $e = r_1 \cdot r_2 \cdots r_n$ ein regulärer Ausdruck, wobei $\cdot$ die Konkatenation bezeichne und $n \geq 1$ ist. Sei $F_i = \text{first}[e_i]$ $(1 \leq i \leq n)$ die Menge der ersten Blätter des regulären Ausdrucks $e_i = r_1 \cdots r_i$. Welche Mengenbeziehung gilt bzgl. der $F_i$? *Hinweis:* Betrachten Sie den passenden Syntaxbaum.

> **Lösung:** $F_i \subseteq F_{i+1}$ $(1 \leq i \leq n-1)$.

(3) Geben Sie eine kontextfreie Grammatik an, die mehr produktive als erreichbare Nichtterminale besitzt.

> **Lösung:** $G = (\{S, A\}, \{a, b\}, \{S \to a, A \to b\}, S)$

(4) Sei $G = (\{S, A, B, C\}, \{a, b, c\}, \{S \to ABC, A \to a, B \to b, C \to BAc\}, S)$ eine kontextfreie Grammatik. Bestimmen Sie die Kardinalität der Übergangsrelation $\delta$ des Shift-Reduce-Parsers $M_G^{(1)}$.

> **Lösung:** Shift-Übergänge: $(4+3+2) \cdot 3 = 27$
> Reduce-Übergänge: $9 \cdot 4 = 36$
> Abschluss: 1
> Ingesamt: $27 + 36 + 1 = 64$

(5) Sei $G$ eine reduzierte kontextfreie Grammatik mit $n$ Terminalsymbolen. Angenommen jede der $n + 1$ Spalten der (starken) Vorausschautabelle hat genau einen Eintrag (eine Produktion). Ist $G$ dann stark $LL(1)$? Begründen Sie Ihre Antwort.

> **Lösung:** Ja, denn wenn in jeder Spalte nur ein Eintrag steht, hat insbesondere jedes Feld höchstens einen Eintrag und damit ist die Grammatik stark $LL(1)$.

**Aufgabe 2.** (12 Punkte) Sei $r \in \mathcal{E}_{\{a,b\}}$ mit $r = ab^* \mid a^+$. Konstruieren Sie den Berry-Sethi-Automaten zu $r$. Geben Sie für jeden Teilausdruck die Werte der Funktionen empty, first, last und next an.

**Lösung:** Durchnummerieren der Terminalzeichen:

$$r' := \mathrm{num}(r) = [1,a][2,b]^* \mid [3,a]^+.$$

$\mathrm{empty}(r') = \mathrm{empty}([1,a][2,b]^*) \vee \mathrm{empty}([3,a]^+) = f$

- $\mathrm{empty}([1,a][2,b]^*) = \mathrm{empty}([1,a]) \wedge \mathrm{empty}([2,b]^*) = f$
  - $\mathrm{empty}([1,a]) = f$
  - $\mathrm{empty}([2,b]^*) = t$
    - $\mathrm{empty}([2,b]) = f$
- $\mathrm{empty}([3,a]^+)) = f$
  - $\mathrm{empty}([3,a])) = f$

$\mathrm{first}(r') = \mathrm{first}([1,a][2,b]^*) \cup \mathrm{first}([3,a]^+) = \{[1,a],[3,a]\}$

- $\mathrm{first}([1,a][2,b]^*) = \mathrm{first}([1,a]) = \{[1,a]\}$
  - $\mathrm{first}([2,b]^*) = \mathrm{first}([2,b]) = \{[2,b]\}$
- $\mathrm{first}([3,a]^+) = \mathrm{first}([3,a]) = \{[3,a]\}$

**Lösung:**

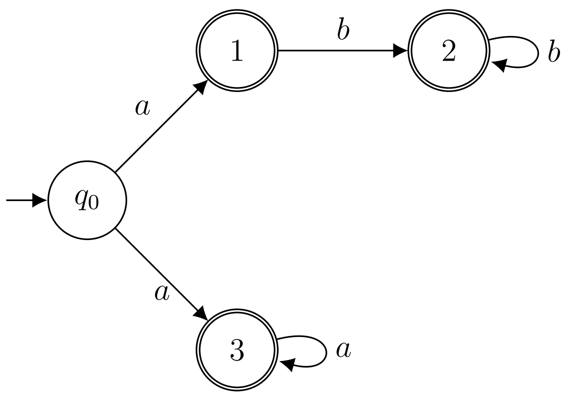
$\text{last}(r') = \text{last}([1,a][2,b]^*) \cup \text{last}([3,a]^+) = \{[1,a],[2,b],[3,a]\}$

- $\text{last}([1,a][2,b]^*) = \text{last}([1,a]) \cup \text{last}([2,b]^*) = \{[1,a],[2,b]\}$
  - $\text{last}([1,a]) = \{[1,a]\}$
  - $\text{last}([2,b]^*) = \text{last}([2,b]) = \{[2,b]\}$
- $\text{last}([3,a]^+) = \text{last}([3,a]) = \{[3,a]\}$

$\text{next}(r') = \emptyset$

- $\text{next}([1,a][2,b]^*) = \text{next}(r') = \emptyset$
  - $\text{next}([1,a]) = \text{first}([2,b]^*) \cup \text{next}([1,a][2,b]^*) = \{[2,b]\}$
  - $\text{next}([2,b]^*) = \text{next}([1,a][2,b]^*) = \emptyset$
    - $\text{next}([2,b]) = \text{first}([2,b]) \cup \text{next}([2,b]^*) = \{[2,b]\}$
- $\text{next}([3,a]^+) = \text{next}(r') = \emptyset$
  - $\text{next}([3,a]) = \text{first}([3,a]) \cup \text{next}(r') = \{[3,a]\}$

Damit ergibt sich der folgende NFA:



**Aufgabe 3.** (12 Punkte) Sei $G = (N, \{[, ], a, *, +\}, P, S)$, wobei $N = \{S, A, B, C\}$ und $P$ gegeben ist durch

$$
\begin{aligned}
S &\to [ABC] \mid \varepsilon \\
A &\to a{*}a \mid B{*}C{+} \\
B &\to {+}aC{*} \\
C &\to [A{*}B] \mid A{*}B
\end{aligned}
$$

(a) Geben Sie $\text{First}_1(X)$ für jedes $X \in N$ an.

**Lösung:**

$$
\begin{aligned}
\text{First}_1(S) &= \{[, \varepsilon\} \\
\text{First}_1(A) &= \{a, +\} \\
\text{First}_1(B) &= \{+\} \\
\text{First}_1(C) &= \{[, a, +\}
\end{aligned}
$$

(b) Geben Sie $\text{Follow}_1(X)$ für jedes $X \in N$ an.

**Lösung:**

$$
\begin{aligned}
\text{Follow}_1(S) &= \{\varepsilon\} \\
\text{Follow}_1(A) &= \{+, *\} \\
\text{Follow}_1(B) &= \{[, a, +, *, ]\} \\
\text{Follow}_1(C) &= \{], +, *\}
\end{aligned}
$$

(c) Geben Sie die Vorausschautabelle für stark LL(1) an.

**Lösung:**

$$
\begin{aligned}
S \to \varepsilon &: \text{First}_1(\varepsilon) \odot_1 \text{Follow}_1(S) = \{\varepsilon\} \\
S \to [ABC] &: \text{First}_1([ABC]) \odot_1 \text{Follow}_1(S) = \{[\} \\
A \to a{*}a &: \text{First}_1(a{*}a) \odot_1 \text{Follow}_1(A) = \{a\} \\
A \to B{*}C{+} &: \text{First}_1(B{*}C{+}) \odot_1 \text{Follow}_1(A) = \{+\} \\
B \to {+}aC{*} &: \text{First}_1({+}aC{*}) \odot_1 \text{Follow}_1(B) = \{+\} \\
C \to [A{*}B] &: \text{First}_1([A{*}B]) \odot_1 \text{Follow}_1(C) = \{[\} \\
C \to A{*}B &: \text{First}_1(A{*}B) \odot_1 \text{Follow}_1(C) = \{a, +\}
\end{aligned}
$$

| | [ | ] | $a$ | $*$ | $+$ | $\varepsilon$ |
|---|---|---|---|---|---|---|
| $S$ | $S \to [ABC]$ | | | | | $S \to \varepsilon$ |
| $A$ | | | $A \to a{*}a$ | | $A \to B{*}C{+}$ | |
| $B$ | | | | | $B \to {+}aC{*}$ | |
| $C$ | $C \to [A{*}B]$ | | $C \to A{*}B$ | | $C \to A{*}B$ | |

**Aufgabe 4.** (6 Punkte) Gegeben sei folgende Übergangsfunktion eines DFAs:

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

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. Wählen Sie den Default-Wert sinnvoll.

**Lösung:** Wir wählen Default $= 4$ und erhalten

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

Dann verschieben wir die Zeile für $b$ um 1 nach rechts und erhalten

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

Damit ergibt sich displacement$(a) = 0$, displacement$(b) = 1$ und

| | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| A | 1 | 2 | 3 | | 1 | 3 |
| valid | a | b | b | | b | a |

**Aufgabe 5.** (10 Punkte) Sei $G = (N, \{a, b, c\}, P, A)$, wobei $N = \{A, B, C\}$ und $P$ gegeben ist durch:

$$\begin{aligned} A &\to cBA \mid ba \\ B &\to Aab \mid BCc \\ C &\to CAb \mid \varepsilon \end{aligned}$$

Berechnen Sie $\text{Follow}_1(X)$ für alle $X \in N$ mit dem Algorithmus aus der Vorlesung. Stellen Sie dazu das Ungleichungssystem auf und vereinfachen Sie dieses so weit wie möglich. Die Werte für $\text{Follow}_1(X)$ nach jedem Schritt dürfen Sie in einer Tabelle festhalten, wobei Sie in der ersten Spalte (Schritt 0) $\text{Follow}_1(A) = \{\varepsilon\}$ und $\text{Follow}_1(B) = \text{Follow}_1(C) = \emptyset$ initialisieren. Ferner dürfen Sie folgenden Informationen benutzen:

$$\begin{aligned} \text{First}_1(A) &= \{b, c\} \\ \text{First}_1(B) &= \{b, c\} \\ \text{First}_1(C) &= \{\varepsilon, b, c\} \end{aligned}$$

**Lösung:** Wir haben das Ungleichungssystem:

$$\begin{aligned} \text{Follow}_1(A) \supseteq \{\varepsilon\} &\cup \text{First}_1(\varepsilon) \odot_1 \text{Follow}_1(A) \\ &\cup \text{First}_1(ab) \odot_1 \text{Follow}_1(B) \\ &\cup \text{First}_1(b) \odot_1 \text{Follow}_1(C) \\ \text{Follow}_1(B) \supseteq &\cup \text{First}_1(A) \odot_1 \text{Follow}_1(A) \\ &\cup \text{First}_1(Cc) \odot_1 \text{Follow}_1(B) \\ \text{Follow}_1(C) \supseteq &\cup \text{First}_1(c) \odot_1 \text{Follow}_1(B) \\ &\cup \text{First}_1(Ab) \odot_1 \text{Follow}_1(C) \end{aligned}$$

Wir vereinfachen das Ganze etwas und erhalten:

$$\begin{aligned} \text{Follow}_1(A) \supseteq \{\varepsilon\} &\cup \{a\} \odot_1 \text{Follow}_1(B) \\ &\cup \{b\} \odot_1 \text{Follow}_1(C) \\ \text{Follow}_1(B) \supseteq &\cup \{b, c\} \odot_1 \text{Follow}_1(A) \\ &\cup \{\varepsilon, b, c\} \odot_1 \{c\} \odot_1 \text{Follow}_1(B) \\ \text{Follow}_1(C) \supseteq &\cup \{c\} \odot_1 \text{Follow}_1(B) \\ &\cup \{b, c\} \odot_1 \{b\} \odot_1 \text{Follow}_1(C) \end{aligned}$$

Damit ergibt sich folgende Tabelle:

| Schritt | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| $\text{Follow}_1(A)$ | $\{\varepsilon\}$ | $\{\varepsilon\}$ | $\{\varepsilon, a\}$ | $\{\varepsilon, a, b\}$ | $\{\varepsilon, a, b\}$ |
| $\text{Follow}_1(B)$ | $\emptyset$ | $\{b, c\}$ | $\{b, c\}$ | $\{b, c\}$ | $\{b, c\}$ |
| $\text{Follow}_1(C)$ | $\emptyset$ | $\emptyset$ | $\{c\}$ | $\{b, c\}$ | $\{b, c\}$ |

Da sich von Schritt 3 auf 4 nichts verändert hat, terminiert der Algorithmus und es gilt $\text{Follow}_1(A) = \{\varepsilon, a, b\}$ und $\text{Follow}_1(B) = \text{Follow}_1(C) = \{b, c\}$.

**Aufgabe 6.** (6 Punkte (Bonus)) Sei $G = (\{S, A, B, C\}, \{a, b, *\}, P, S)$, wobei $P$ gegeben ist durch:

$$
\begin{aligned}
S &\to aaBb \mid abAb \\
A &\to aB \mid C \\
B &\to bA \mid C \\
C &\to *abba \mid CS
\end{aligned}
$$

Geben Sie eine LL(1)-Grammatik $G'$ an mit $L(G') = L(G)$.

**Lösung:** $G' = (\{S, A, B, C, D, E\}, \{a, b, *\}, P', S)$, wobei $P'$ gegeben ist durch:

$$
\begin{aligned}
S &\to aD \\
D &\to aBb \mid bAb \\
A &\to aB \mid C \\
B &\to bA \mid C \\
C &\to *abbaE \\
E &\to \varepsilon \mid SE
\end{aligned}
$$