Prüfung zur Vorlesung „Compilerbau“

SS 2021 / 12. August 2021

Vorname: ____________________

Nachname: ____________________

Matrikelnummer: ____________________

Aufgabe	Punktzahl	Erreicht
1	14	
2	4	
3	12	
4	10	
5	10	
6	0	
Σ	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.
- Alternativ können Sie die Lösungen auch direkt auf einem Tablet mit der Hand schreiben und uns das PDF schicken.
- 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 12. August 2021 (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 folgenden Telefonnummern erreichen:

  [redacted]

  Wir können nicht garantieren, dass diese Nummern auch stets erreichbar sind.

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. (14 Punkte) Sei $r \in \mathcal{E}_{\{a,b,c\}}$ mit $r = (ca)^*b^+ \,|\, a$. 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:

$$r' := \mathrm{num}(r) = ([1,c][2,a])^*[3,b]^+ \mid [4,a].$$

$\mathrm{empty}(r') = \mathrm{empty}(([1,c][2,a])^*[3,b]^+) \vee \mathrm{empty}([4,a]) = f$
- $\mathrm{empty}(([1,c][2,a])^*[3,b]^+) = \mathrm{empty}(([1,c][2,a])^*) \wedge \mathrm{empty}([3,b]^+) = f$
  - $\mathrm{empty}(([1,c][2,a])^*) = t$
    - $\mathrm{empty}(([1,c][2,a])) = \mathrm{empty}([1,c]) \wedge \mathrm{empty}([2,a]) = f$
      - $\mathrm{empty}([1,c]) = f$
      - $\mathrm{empty}([2,a]) = f$
  - $\mathrm{empty}([3,b]^+) = f$
    - $\mathrm{empty}([3,b]) = f$
- $\mathrm{empty}([4,a]) = f$

$\mathrm{first}(r') = \mathrm{first}(([1,c][2,a])^*[3,b]^+) \cup \mathrm{first}([4,a]) = \{[1,c],[3,b],[4,a]\}$
- $\mathrm{first}(([1,c][2,a])^*[3,b]^+) = \mathrm{first}(([1,c][2,a])^*) \cup \mathrm{first}([3,b]^+) = \{[1,c],[3,b]\}$
  - $\mathrm{first}(([1,c][2,a])^*) = \mathrm{first}([1,c][2,a]) = \{[1,c]\}$
    - $\mathrm{first}([1,c][2,a]) = \mathrm{first}([1,c]) = \{[1,c]\}$
      - $\mathrm{first}([1,c]) = \{[1,c]\}$
      - $\mathrm{first}([2,a]) = \{[2,a]\}$
  - $\mathrm{first}([3,b]^+) = \mathrm{first}([3,b]) = \{[3,b]\}$
    - $\mathrm{first}([3,b]) = \{[3,b]\}$
- $\mathrm{first}([4,a]) = \{[4,a]\}$

Lösung:

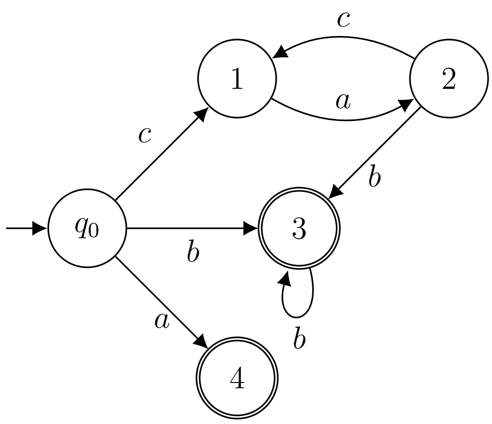
$\text{last}(r') = \text{last}(([1,c][2,a])^*[3,b]^+) \cup \text{last}([4,a]) = \{[3,b],[4,a]\}$

- $\text{last}(([1,c][2,a])^*[3,b]^+) = \text{last}([3,b]^+) = \{[3,b]\}$
  - $\text{last}(([1,c][2,a])^*) = \text{last}([1,c][2,a]) = \{[2,a]\}$
    - $\text{last}([1,c][2,a]) = \text{last}([2,a]) = \{[2,a]\}$
      - $\text{last}([1,c]) = \{[1,c]\}$
      - $\text{last}([2,a]) = \{[2,a]\}$
  - $\text{last}([3,b]^+) = \text{last}([3,b]) = \{[3,b]\}$
    - $\text{last}([3,b]) = \{[3,b]\}$
- $\text{last}([4,a]) = \{[4,a]\}$

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

- $\text{next}(([1,c][2,a])^*[3,b]^+) = \text{next}(r') = \emptyset$
  - $\text{next}(([1,c][2,a])^*) = \text{first}([3,b]^+) = \{[3,b]\}$
    - $\text{next}([1,c][2,a]) = \text{first}([1,c][2,a]) \cup \text{next}(([1,c][2,a])^*) = \{[1,c],[3,b]\}$
      - $\text{next}([1,c]) = \text{first}([2,a]) = \{[2,a]\}$
      - $\text{next}([2,a]) = \text{next}([1,c][2,a]) = \{[1,c],[3,b]\}$
  - $\text{next}([3,b]^+) = \text{next}(([1,c][2,a])^*[3,b]^+) = \emptyset$
    - $\text{next}([3,b]) = \text{next}([3,b]^+) \cup \text{first}([3,b]) = \{[3,b]\}$
- $\text{next}([4,a]) = \text{next}(r') = \emptyset$

Damit ergibt sich der folgende NFA:

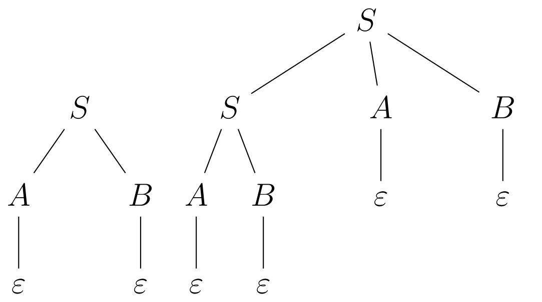


Aufgabe 2. (4 Punkte) Sei $G = (\{S, A, B\}, \{a, b, c\}, P, S)$ eine kontextfreie Grammatik, wobei P gegeben ist durch;

$$\begin{aligned} S &\to AB \mid SAB \\ A &\to Aa \mid \varepsilon \mid ab \\ B &\to \varepsilon \mid bc \mid Bb \end{aligned}$$

Zeigen Sie, dass G nicht eindeutig ist, d.h. finden Sie ein Wort, das mindestens zwei verschiedene Ableitungsbäume hat.

Lösung: Nehme $w = \varepsilon$:



Aufgabe 3. (12 Punkte) Sei $G = (N, \{(,), +, -, \text{int}\}, P, S)$, wobei $N = \{S, A, B\}$ und P gegeben ist durch

$$\begin{aligned} S &\to (A{+}B) \mid \varepsilon \\ A &\to +AB \mid \text{int} \\ B &\to -\text{int}{+}\text{int} \mid S{+}\text{int} \end{aligned}$$

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

Lösung:

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

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

Lösung:

$$\begin{aligned} \text{Follow}_1(S) &= \{\varepsilon, +\} \\ \text{Follow}_1(A) &= \{+, -, (\} \\ \text{Follow}_1(B) &= \{+, -, (,)\} \end{aligned}$$

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

Lösung:

$$\begin{aligned} S \to \varepsilon &: \text{First}_1(\varepsilon) \odot_1 \text{Follow}_1(S) = \{\varepsilon, +\} \\ S \to (A{+}B) &: \text{First}_1((A{+}B)) \odot_1 \text{Follow}_1(S) = \{(\} \\ A \to +AB &: \text{First}_1(+AB) \odot_1 \text{Follow}_1(A) = \{+\} \\ A \to \text{int} &: \text{First}_1(\text{int}) \odot_1 \text{Follow}_1(A) = \{\text{int}\} \\ B \to -\text{int}{+}\text{int} &: \text{First}_1(-\text{int}{+}\text{int}) \odot_1 \text{Follow}_1(B) = \{-\} \\ B \to S{+}\text{int} &: \text{First}_1(S{+}\text{int}) \odot_1 \text{Follow}_1(B) = \{(, +\} \end{aligned}$$

	(	)	+	−	int	ε
S	$S \to (A{+}B)$		$S \to \varepsilon$			$S \to \varepsilon$
A			$A \to +AB$		$A \to \text{int}$	
B	$B \to S{+}\text{int}$		$B \to S{+}\text{int}$	$B \to -\text{int}{+}\text{int}$		

Aufgabe 4. (10 Punkte) Sei $G = (\{S, A, B\}, \{a, b, +, *\}, P, S)$, wobei P gegeben ist durch:

$$
\begin{aligned}
S &\to AB+ \mid AB* \\
A &\to aA \mid \varepsilon \\
B &\to bB \mid \varepsilon
\end{aligned}
$$

(a) Zeigen Sie, dass $G \notin \mathrm{LL}(k)$ für alle $k \in \mathbb{N}$.

> **Lösung:** Sei $k \in \mathbb{N}$. Betrachte die Linksableitung $S \to_L^* S$ und die beiden Produktionen $S \to AB+$ und $S \to AB*$. Wir sehen, dass Wörter der Form $w_k = a^k+$ aus der ersten und Wörter der Form $v_k = a^k*$ aus der zweiten Produktion ableitbar sind. Somit enthält der Schnitt
>
> $$\mathrm{First}_k(AB+) \cap \mathrm{First}_k(AB*)$$
>
> stets das Wort $\mathrm{First}_k(w_k) = \mathrm{First}_k(v_k) = a^k$. Folglich ist G keine LL(k)-Grammatik.

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

> **Lösung:** $G' = (\{S, A, B\}, \{a, b, +, *\}, P', S)$, wobei P' gegeben ist durch:
>
> $$
> \begin{aligned}
> S &\to aS \mid A \\
> A &\to bA \mid B \\
> B &\to + \mid *
> \end{aligned}
> $$

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

$$A \to aba \mid bB$$
$$B \to bb \mid AB$$

Berechnen Sie $\mathrm{First}_2(X)$ für alle $X \in N$ mit dem Algorithmus aus der Vorlesung.

Lösung: Wir haben das Ungleichungssystem

$$\mathrm{First}_2(A) \supseteq \mathrm{First}_2(aba) \cup (\mathrm{First}_2(b) \odot_2 \mathrm{First}_2(B))$$
$$\mathrm{First}_2(B) \supseteq \mathrm{First}_2(bb) \cup (\mathrm{First}_2(A) \odot_2 \mathrm{First}_2(B))$$

Initialisiere $\mathrm{First}_2(A) \supseteq \emptyset$ und $\mathrm{First}_2(B) \supseteq \emptyset$. Wir erhalten:

$$\mathrm{First}_2(A) \supseteq \{ab\} \cup (\{b\} \odot_2 \emptyset) = \{ab\}$$
$$\mathrm{First}_2(B) \supseteq \{bb\} \cup (\emptyset \odot_2 \emptyset) = \{bb\}$$

Also haben wir $\mathrm{First}_2(A) \supseteq \{ab\}$ und $\mathrm{First}_2(B) \supseteq \{bb\}$ und erhalten:

$$\mathrm{First}_2(A) \supseteq \{ab\} \cup (\{b\} \odot_2 \{bb\}) = \{ab, bb\}$$
$$\mathrm{First}_2(B) \supseteq \{bb\} \cup (\{ab\} \odot_2 \{bb\}) = \{ab, bb\}$$

Also haben wir $\mathrm{First}_2(A) \supseteq \{ab, bb\}$ und $\mathrm{First}_2(B) \supseteq \{ab, bb\}$ und erhalten:

$$\mathrm{First}_2(A) \supseteq \{ab\} \cup (\{b\} \odot_2 \{ab, bb\}) = \{ab, ba, bb\}$$
$$\mathrm{First}_2(B) \supseteq \{bb\} \cup (\{ab, bb\} \odot_2 \{ab, bb\}) = \{ab, bb\}$$

Also haben wir $\mathrm{First}_2(A) \supseteq \{ab, ba, bb\}$ und $\mathrm{First}_2(B) \supseteq \{ab, bb\}$ und erhalten:

$$\mathrm{First}_2(A) \supseteq \{ab\} \cup (\{b\} \odot_2 \{ab, bb\}) = \{ab, ba, bb\}$$
$$\mathrm{First}_2(B) \supseteq \{bb\} \cup (\{ab, ba, bb\} \odot_2 \{ab, bb\}) = \{ab, ba, bb\}$$

Also haben wir $\mathrm{First}_2(A) \supseteq \{ab, ba, bb\}$ und $\mathrm{First}_2(B) \supseteq \{ab, ba, bb\}$ und erhalten:

$$\mathrm{First}_2(A) \supseteq \{ab\} \cup (\{b\} \odot_2 \{ab, ba, bb\}) = \{ab, ba, bb\}$$
$$\mathrm{First}_2(B) \supseteq \{bb\} \cup (\{ab, ba, bb\} \odot_2 \{ab, ba, bb\}) = \{ab, ba, bb\}$$

Da sich nichts verändert hat, terminiert der Algorithmus und es gilt $\mathrm{First}_2(A) = \{ab, ba, bb\}$ und $\mathrm{First}_2(B) = \{ab, ba, bb\}$

Aufgabe 6. (8 Punkte (Bonus)) Sei $G = (N, \{+, :, ?, \langle, \rangle\}, P, S)$ eine reduzierte kontextfreie Grammatik. Sei ferner $A \to +B\colon \mid C \mid D? \in P$, wobei $A, B, C, D \in N$. Es gelten weiter $|\,\mathrm{Follow}_1(A)| \geq 4$, $|\,\mathrm{First}_1(D)| \geq 3$ und $\varepsilon \in \mathrm{First}_1(C)$. Zeigen Sie: G ist nicht stark LL(1).

Hinweis: Überlegen Sie sich, wie die Zeile von A in der Vorausschautabelle aussehen muss.

Lösung: Die Vorausschautabelle für stark LL(1) dieser Grammatik hat genau 6 Spalten (Anzahl der Terminalsymbole plus eins für ε). Wir werden via Schubfachprinzip zeigen, dass mindestens eine Spalte in der Zeile mit dem Symbol A mit zwei verschiedenen Regeln $A \to \alpha$ ausgefüllt werden muss. Damit wäre die Grammatik nicht stark LL(1).

Wir schätzen zunächst die Anzahl der Elemente in $\mathrm{First}_1(\alpha) \odot_1 \mathrm{Follow}_1(A)$ nach unten ab:

$$\begin{aligned}
|\,\mathrm{First}_1(+B\colon) \odot_1 \mathrm{Follow}_1(A)| &= |\{+\}| = 1 \\
|\,\mathrm{First}_1(C) \odot_1 \mathrm{Follow}_1(A)| \geq |\{\varepsilon\} \odot_1 \mathrm{Follow}_1(A)| = |\,\mathrm{Follow}_1(A)| &\geq 4 \\
|\,\mathrm{First}_1(D?) \odot_1 \mathrm{Follow}_1(A)| &\geq 2
\end{aligned}$$

Für die letzte Ungleichung müssen wir 2 Fälle betrachten: Falls ε und ? in $\mathrm{First}_1(D)$ enthalten sind, sind nur mindestens 2 Elemente in $\mathrm{First}_1(D?)$ enthalten (? und ein weiteres Terminalzeichen). Ansonsten hat $\mathrm{First}_1(D?)$ sogar mindestens 3 Elemente.

Insgesamt erhalten wir also 7 Symbole für die 3 Produktionen, also müssen 7 Regeln auf 6 Spalten aufgeteilt werden. Nach dem Schubfachprinzip stehen mindestens 2 Regeln in einer Zelle der Tabelle, was zu zeigen war.