

Compilerbau - Wintersemester 2021/22

Übungsblatt 7 - Musterlösung

Aufgabe 7.1

(a) Es gibt zwei Linksableitungen für $a|a|a$, nämlich

$$R \rightarrow R|R \rightarrow R|R|R \rightarrow a|R|R \rightarrow a|a|R \rightarrow a|a|a$$

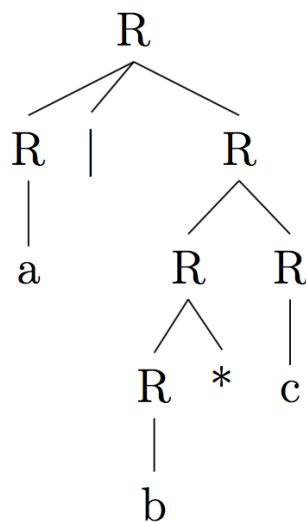
und

$$R \rightarrow R|R \rightarrow a|R \rightarrow a|R|R \rightarrow a|a|R \rightarrow a|a|a.$$

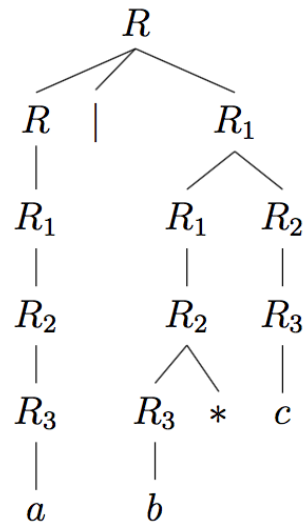
(b)

$$\begin{aligned} R &\rightarrow R|R_1 \mid R_1 \\ R_1 &\rightarrow R_1R_2 \mid R_2 \\ R_2 &\rightarrow R_3* \mid R_3 \\ R_3 &\rightarrow (R) \mid a \mid b \mid c \end{aligned}$$

(c) Ein Syntaxbaum, der den Operatorpräzendenzen entspricht wäre:

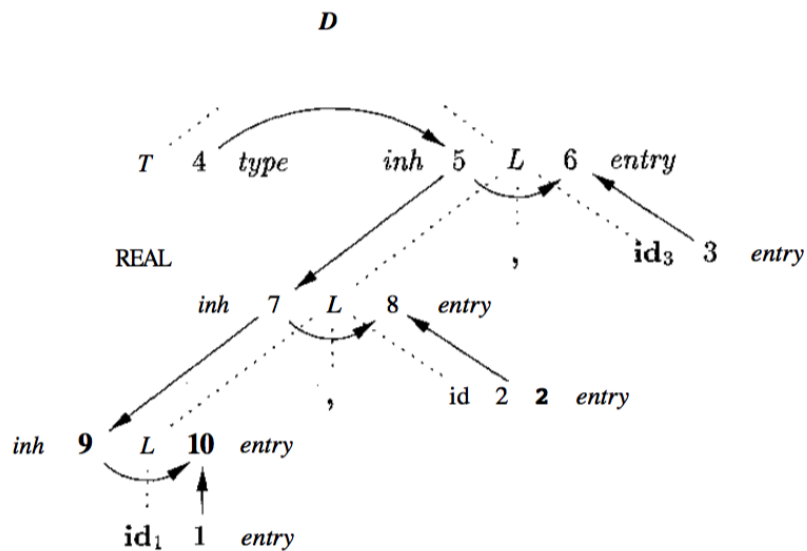


Für die Grammatik aus (b) ist der Syntaxbaum:



Aufgabe 7.2

Hinweis: Zu L wird das Pseudoattribut entry eingefügt um die Anwendung der Funktion addType zu verdeutlichen.



Aufgabe 7.3

Es wurden aus Gründen der Übersichtlichkeit Indizes hinzugefügt. Die (mehrdeutige) S-Attributgrammatik ist:

$$S \rightarrow head\ begin\ Statement\ end.$$

$Statement \rightarrow Statement; Statement$

$Statement \rightarrow id = E; \{symboltable.put(id.lexvalue, E.value)\}$

$Statement \rightarrow writeln(E); \{println(E.value)\}$

$E_0 \rightarrow E_1 \text{ or } E_2 \{E_0.value = E_1.value \text{ or } E_2.value\}$

$E_0 \rightarrow E_1 \text{ and } E_2 \{E_0.value = E_1.value \text{ and } E_2.value\}$

$E_0 \rightarrow \text{not } E_1 \{E_0.value = \text{not } E_1.value\}$

$E \rightarrow \text{true} \{E.value = \text{true}\}$

$E \rightarrow \text{false} \{E.value = \text{false}\}$

$E \rightarrow id \{E.value = symboltable.lookup(id.lexvalue)\}$

Aufgabe 7.4

Alle Knoten erben vom Knoten R. Die S-Attributgrammatik ist:

$R_0 \rightarrow R_0 \text{ '}' R_1 \{R_0.ast = new OrNode(R_0.ast, R_1.ast)\}$

$R_0 \rightarrow R_1 \{R_0.ast = R_1.ast\}$

$R_1 \rightarrow R_1 R_2 \{R_2.ast = new ConcNode(R_1.ast, R_2.ast)\}$

$R_1 \rightarrow R_2 \{R_1.ast = R_2.ast\}$

$R_2 \rightarrow R_2 * \{R_2.ast = new StarNode(R_2.ast)\}$

$R_2 \rightarrow (R_0) \{R_2.ast = R_0.ast\}$

$R_2 \rightarrow a \{R_2.ast = new AtomNode(a)\}$

$R_2 \rightarrow b \{R_2.ast = new AtomNode(b)\}$

$R_2 \rightarrow c \{R_2.ast = new AtomNode(c)\}$

