

Compilerbau
8. Übungsblatt SS 08
Abgabetermin: 10.06.08

Aufgabe 28 (5 Punkte)

Gegeben sei die folgende Grammatik für reguläre Ausdrücke

$$\begin{aligned} E &\rightarrow E + T \mid T \\ T &\rightarrow T F \mid F \\ F &\rightarrow (E) \mid \text{id} \mid F * \end{aligned}$$

Die Terminalzeichen sind +, (,), id und *.

- Eliminieren Sie mit dem Verfahren aus der Vorlesung die Linksrekursion.
- Berechnen Sie die Start- und Folgemengen und erstellen Sie damit die Parse-Tabelle.
- Veranschaulichen Sie für die Eingaben „(id + id) *“ und „id id (*“ die Aktionen des Parsers mittels der Darstellung des Kellerinhaltes und des noch nicht gelesenen Teils des Eingabewortes.

Aufgabe 29

Gegeben sei die erweiterte kontextfreie Grammatik $G = (N, T, S', P)$ mit $N = \{S', S, A\}$, $T = \{a, b, c, \$\}$ und den Produktionen

$$\begin{aligned} S' &\rightarrow S\$ \\ S &\rightarrow Sb \mid bAa \\ A &\rightarrow aSc \mid aSb \mid a \end{aligned}$$

Berechnen Sie die zugehörigen Linkskontexte und die Menge der maximalen zuverlässigen Präfixe pre_G .

Aufgabe 30

Erstellen Sie eine Tabelle mit einer sinnvollen Prioritätsrelation für die Operatoren der folgenden Grammatik:

$$\begin{aligned} E &\rightarrow E+E \mid E * E \mid !E \mid \&E \mid \text{id} \mid \text{id}[E] \mid D \\ D &\rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \end{aligned}$$

Hierbei bezeichnet ! den unären not-Operator in C und & den unären Adress-Operator in C. Demonstrieren Sie die Schritte des Operator-Präzedenz-Parsers anhand der Eingaben „values[1+n*m]“ und „!&reference“.

Aufgabe 31

Konstruieren Sie einen Operator-Präzedenz-Parser für reguläre Ausdrücke mit korrekter Klammerung.