Lehrstuhl für Programmiersprachen und Übersetzerkonstruktion
CAU Kiel, Informatik und Prakt. Mathematik
Prof. Dr. M. Hanus, Frank Huch

24118 Kiel
Christan-Albrechts-Platz 4
Tel.: 0431 / 880-7277

5. Übung „Übersetzerkonstruktion“

Abgabe am 19. November in der Vorlesung

Aufgabe 15 5 Punkte

In der Vorlesung wurde die LL(k)- und die starke LL(k)(SLL(k))-Eigenschaft definiert. Zeigen Sie anhand folgender Grammatik, daß SLL(2) $\neq$ LL(2) gilt:

$$\begin{aligned} S &\rightarrow aAab \mid bAbb \\ A &\rightarrow a \mid \varepsilon \end{aligned}$$

Bem.: Für $k = 1$ stimmen beide Eigenschaften überein.

Aufgabe 16 4 Punkte

a) Geben Sie eine zur Definition äquivalente Charakterisierung der Menge der LL(0)–Grammatiken sowie der Menge der LL(0)–Sprachen (d.h. der durch LL(0)–Grammatiken erzeugbaren Sprachen) an. Diese Definition soll natürlich einfacher sein, als die in der Vorlesung gegebene.

b) Zeigen Sie, daß jede reguläre Sprache durch eine LL(1)–Grammatik erzeugt wird.

Aufgabe 17 3 Punkte

In der Vorlesung wurde die Linksfaktorisierung am Beispiel des sogenannten „dangling if-then-else“ vorgestellt. Überprüfen Sie, ob die resultierende Grammatik die LL(1)-Eigenschaft besitzt. Können Sie diese Grammatik ggf. durch weitere Linksfaktorisierungen bzw. durch Elimination von Linksrekursion in eine LL(1)-Grammatik überführen?

Aufgabe 18 8 Punkte

Die Syntax der Mini–Programmiersprache MPS ist durch unten folgende Syntaxdiagramme gegeben.

a) Geben Sie die Syntax von MPS durch eine kontextfreie Grammatik G an.

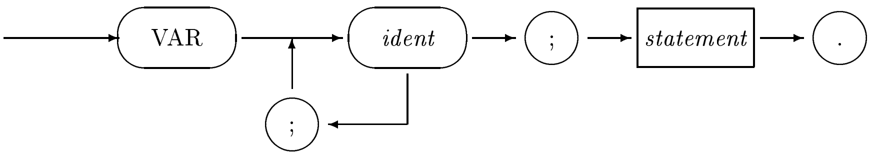
b) Berechnen Sie zu allen Nichtterminalsymbolen aus G die Mengen FIRST und FOLLOW.

c) Prüfen Sie, ob es sich bei Ihrer Grammatik um eine LL(1)–Grammatik handelt. Wenn nicht, so geben Sie eine LL(1)–Grammatik G' für MPS an.

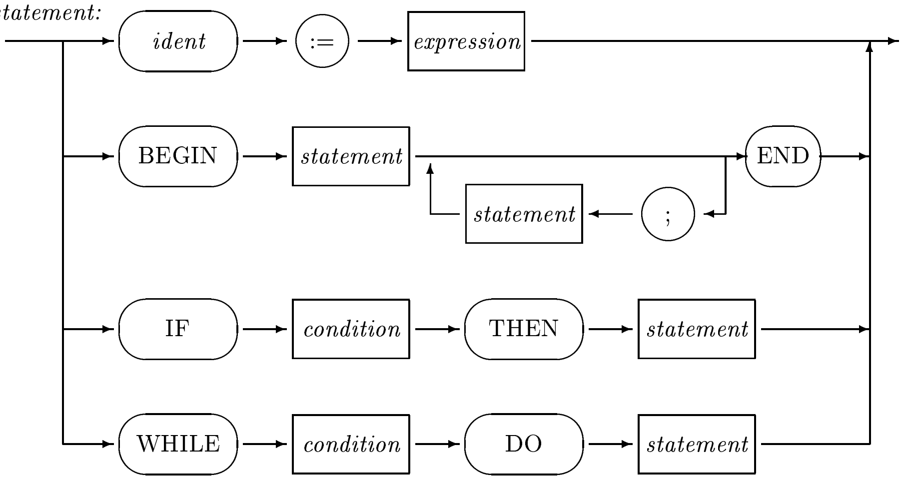
d) Konstruieren Sie die zugehörige Parsing-Tabelle.

e) Für die Implementierung des Parsers modifizieren wir MPS wie folgt: statt durch Semikolons werden Bezeichner bei ihrer Deklaration durch Kommas getrennt. Implementieren Sie den rekursiven Abstiegsparser für MPS in Haskell unter Berücksichtigung der (angepaßten) Parsing-Tabelle.

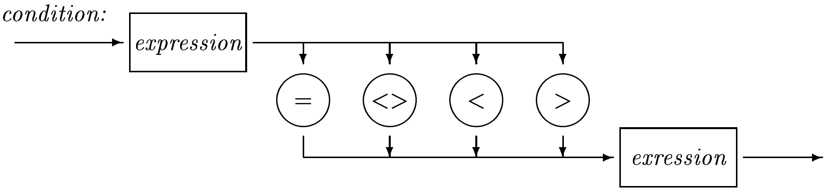
program:



statement:



condition:



expression: