

Compilerbau<http://proglang.informatik.uni-freiburg.de/teaching/compilerbau/2004/>

Übungsblatt 3

Abgabe: 17.11.2004

Aufgabe 1 (LL(0)-Sprachen):

Eine Sprache ist LL(k)-Sprache, falls es eine LL(k)-Grammatik für sie gibt.

Welche Sprachen gehören zu der Klasse der LL(0)-Sprachen? Beweise Deine Vermutung.

Aufgabe 2 (LL(1)-Sprachen):

Zeige, dass die Grammatik $G = (\{S', S, B, E, J, L\}, \{;, :=, (,), ,\}, P, S')$ mit Produktionen P

$$\begin{aligned} S' &\rightarrow S \\ S &\rightarrow LB \\ B &\rightarrow ;S \mid :=L \\ E &\rightarrow a \mid L \\ J &\rightarrow ,EJ \mid) \\ L &\rightarrow (EJ \end{aligned}$$

LL(1) ist.

Aufgabe 3 (*recursive descent*-Parser):

Implementiere einen *recursive descent*-Parser, der Wörter für die durch die Grammatik $G = (\{S, B, T, T', E, E', F, N, N'I, D\}, \{;, :=, +, *, (,), a, \dots, z, 0, \dots, 9\}, P, S)$ erzeugte Sprache (bestehend aus einfachen mathematische Ausdrücke mit Deklarationen) mit Produktionen P

$$\begin{aligned} S &\rightarrow \epsilon \mid B;S \mid T \\ B &\rightarrow I:=T \\ T &\rightarrow ET' \\ T' &\rightarrow \epsilon \mid +ET' \\ E &\rightarrow FE' \\ E' &\rightarrow \epsilon \mid *FE' \\ F &\rightarrow I \mid N \mid (T) \\ N &\rightarrow DN' \\ N' &\rightarrow \epsilon \mid DN' \\ I &\rightarrow a \mid \dots \mid z \\ D &\rightarrow 0 \mid \dots \mid 9 \end{aligned}$$

erkennt.

Ein lexikalisch gültiges Programm dieser Sprache ist etwa:

```
a:=23+4*6-5;
b:=(10+10)*2+2;
c:=a-b;
c*c
```

Faktorisiere die Grammatik gegebenenfalls und/oder suche einen passenden Lookahead k , damit ein *recursive descent*-Parser direkt implementiert werden kann.