

Compilerbau WS 09/10

Hausaufgabe 9

1. Gegeben sei folgende Grammatik:

- Startsymbol E
- Terminale +, *, n, (,)
- $E ::= T E'$
- $E' ::= + T E' \mid \varepsilon$
- $T ::= F T'$
- $T' ::= * F T' \mid \varepsilon$
- $F ::= n \mid '(E)'$

- a) Berechne die First- und Follow-Mengen aller Nichtterminale (E, E', T, T', F).
- b) Leite aus den First- und Follow-Mengen die Parsertabelle M der Grammatik ab.
- c) Gegeben sei folgender Eingabetext:

$(4 + 2) * 3$

Wandle zuerst die Eingabe in Parsertoken (Terminale) um. Das Terminalsymbol n entspricht den Ziffern, Whitespace kann ignoriert werden. Verwende anschließend die Parsertabelle zusammen mit dem Algorithmus aus der Vorlesung, um die Eingabe zu parsen. Gib den Stack, das aktuelle Token und die angewandten Regeln für jeden Schritt an.

2. Die Grammatik aus Aufgabe 1 soll nun um Zuweisungen erweitert werden. Dazu wird eine neue Grammatikregel hinzugefügt:

- $E ::= n '=' T E'$

- a) Welche der in 1.a) berechneten Mengen ändern sich? Berechne ggf. ihren neuen Inhalt.
- b) Welche Einträge in der Parsertabelle ändern sich? Gib diese an.
- c) Ist der folgende Ausdruck ein gültiges Wort der Sprache?

$6 = 4 * 2$

Wenn ja, warum lässt er sich mit dem in der Vorlesung gezeigten Algorithmus nicht erfolgreich parsen?

Abgabe **handschriftlich** bis zum Freitag, dem 29. Januar in den Übungsschrank im Fachgebiet SE oder direkt vor der Vorlesung. Bitte Namen und Matrikelnummer angeben!