

# Praktikum 2 Compilerbau WS14/15 Testat bis 04.11.2014

**Ziele:**

- Transformation der Infix-Ausdrücke aus Versuch 1 in Postfixnotation.
- Vorbereitung der Ausdrucksübersetzung.

**Aufgabe: (Postfixtransformation)**

Erweitern Sie den Parser aus Versuch 1, so dass die dort definierten Ausdrücke in Postfixnotation (ohne Klammern) transformiert werden.

Bsp.: Der Ausdruck  $3 * (5 + 17)$  wird transformiert in  $3 5 17 + * .$   
Der Ausdruck  $3 * 5 + 17$  wird transformiert in  $3 5 * 17 + .$   
Der Ausdruck  $3 + 5 * 17$  wird transformiert in  $3 5 17 * + .$   
Der Ausdruck  $8 / 2 / 2$  wird transformiert in  $8 2 / 2 / .$

Dabei sollen die Zahlen und die Operanden jeweils durch ein Leerzeichen getrennt werden und die üblichen Präferenzregeln angewendet werden.

Erweitern Sie die Syntaxregeln ggf. so, dass Regeln zur Priorität geklammerter Ausdrücke und Punkt-vor-Strichrechnung berücksichtigt werden.

**Hinweise:**

Führen Sie die Transformation unmittelbar während des Parsens durch, indem Sie Java-Code zwischen die Komponenten der rechten Regelseiten schreiben.

Siehe hierzu die Erläuterung in der Einführungsvorlesung.

Führen Sie, falls notwendig, Rückgabetypen der einzelnen Methoden ein (z.B. String) und sammeln Sie darin die bereits erzeugten Ausgaben.

Zur Darstellung des aktuellen Tokens kann man das Attribut `token.image` verwenden.

Die Präferenzregeln „Punktrechnung vor Strichrechnung“ und Klammern zuerst erreicht man durch die Wahl einer geschickten Grammatik. Siehe hierzu Syntax von Mini-Java.

Schauen Sie sich die zu JavaCC gehörigen Beispiele an!