



# Universität Karlsruhe (TH)

## Lehrstuhl für Programmierparadigmen

Compilerpraktikum SS 2010

Dozent: Prof. Dr.-Ing. G. Snelting

Betreuer: Matthias Braun

Betreuer: Sebastian Buchwald

<http://pp.info.uni-karlsruhe.de/>

[snelting@ipd.info.uni-karlsruhe.de](mailto:snelting@ipd.info.uni-karlsruhe.de)

[matthias.braun@kit.edu](mailto:matthias.braun@kit.edu)

[sebastian.buchwald@kit.edu](mailto:sebastian.buchwald@kit.edu)

Übungsblatt 5

Ausgabe: 12.05.2010

Besprechung: 19.05.2010

Ziel in dieser Woche ist die Entwicklung der semantische Prüfung. Die semantische Analyse holt Dinge nach, die wegen der beschränkten Grammatikklassen nicht während des Parsens erledigt werden können. Dazu gehören Dinge wie Namensanalyse und Typprüfung.

Im allgemeinen gilt: Liefert diese Prüfung keine Fehler so muss der Compiler (bis auf Resourcebeschränkungen) in der Lage sein das Programm zu übersetzen.

### Aufgabe 1: Namensanalyse

- Welche Elemente im AST müssen verlinkt werden? Welche Namensräume gibt es? Wie sind die Namensräume verschachtelt?
- Entwerft eine Schnittstelle zur Namensanalyse während eines AST-Durchlaufs und implementiert diese.
- Benutzt eure Namensanalyse um Deklarationen mit ihren Benutzern zu verbinden und legt die Ergebnisse im AST ab.

(Wir empfehlen sich die Vorlesungsfolien und Übungsblatt 8 der Vorlesung „Sprachtechnologie und Compiler“ aus dem Wintersemester 2009 anzusehen).

### Aufgabe 2: Semantische Prüfung - Typen

- Welche Typen existieren in MiniJava? Braucht man weitere Typen?
- Stellen Sie Typisierungsregeln für die Operationen in MiniJava auf.
- Implementieren Sie diese Typisierungsregeln.

### Aufgabe 3: Semantische Prüfung - Sonstiges

- Welche weiteren Tests sind in der Semantikphase nötig?
- Implementieren Sie diese!

### Aufgabe 4: Testen

Implementieren Sie einen Schalter `--check` in ihrem Compiler der die angegebene Datei parst und anschließend die semantische Analyse durchführt. Fehlerhafte Programme sollten abgelehnt werden, der Compiler sollte dazu eine Meldung Ausgaben in der die Zeichenkette `error` vorkommt und das Programme sollte mit einem Rückgabewert ungleich 0 beendet werden (`System.exit(1);`).

Die fertigen Compiler die `--print-ast` (von Blatt4) und `--check` implementieren sollten Sie bei uns bis 25.5. abgeben.

Tauschen Sie ihre Testprogramme über das Wiki mit den anderen Gruppen aus:

[http://pp.info.uni-karlsruhe.de/lehre/SS2010/comprakt/wiki/index.php/Testprogramme\\_Semantik](http://pp.info.uni-karlsruhe.de/lehre/SS2010/comprakt/wiki/index.php/Testprogramme_Semantik)