

# 5 Abstrakter Syntaxbaum

5.1 Einleitung

5.2 Attributierte Grammatiken

5.3 Erzeugung des AST für SPL

5.4 Zusammenfassung

5.5 Übungen

5.5.1 Erweiterungen

5.5.2 ANTLR

# 6 Semantische Analyse

6.1 Einleitung

## 6.2 Namensanalyse

6.2.1 Symboltabellen

6.2.2 Das Visitor-Pattern

6.2.3 Typdeklarationen

6.2.4 Variablendeklarationen

6.2.5 Prozedurdeklarationen

## 6.3 Typanalyse

6.3.1 Typanalyse für Ausdrücke

6.3.2 Typanalyse für Anweisungen

## 6.4 Semantische Analyse komplett

## 6.5 Vorgehen

## 6.6 Zusammenfassung

## 6.7 Übungen

6.7.1 Typen

6.7.2 Symboltabelle

6.7.3 Typanalyse

# 7 Variablenallokation

---

## 7.1 Einleitung

## 7.2 Aktivierungsrahmen

7.2.1 Aufrufargumente

7.2.2 Lokale Variablen

7.2.3 Sichern der Register

7.2.4 Beispiel für Speicherallokation

## 7.3 Umsetzung im SPL-Compiler

## 7.4 Dynamische

## Speicherverwaltung

7.4.1 Explizite Deallokation

7.4.2 Implizite Deallokation

## 7.5 Erweiterungen für andere Sprachen

7.5.1 Zugriff auf Variablen eines umgebenden Gültigkeitsbereichs

7.5.2 Funktionen

7.5.3 Weitere Datentypen

7.6 Zusammenfassung

7.7 Übungen

7.7.1 AllocatorVisitor

7.7.2 Aktivierungsrahmen

7.7.3 Implementierung

## 8 Codegenerierung

---

8.1 Einleitung

8.2 Ziel-Hardware

8.2.1 RISC versus CISC

8.3 ECO32

- 8.3.1 Unbedingte Sprungbefehle
- 8.3.2 Befehle zum Speicherzugriff
- 8.3.3 Rechenbefehle
- 8.3.4 Sprungmarken (Labels)
- 8.3.5 Bedingte Sprünge

## 8.4 Codemuster

- 8.4.1 Ausdrücke
- 8.4.2 Zuweisungen
- 8.4.3 If-Anweisung
- 8.4.4 While-Schleifen
- 8.4.5 Zusammengesetzte Anweisung
- 8.4.6 Prozeduren
- 8.4.7 Prozeduraufrufe
- 8.4.8 Beispiel
- 8.4.9 Andere Anweisungstypen
- 8.4.10 Assembler-Direktiven
- 8.4.11 Post-Processing

## 8.5 Umsetzung im SPL-Compiler