

# Compilerbau

- Ziel
- Inhalte
- Voraussetzung
- Werkzeuge
- Literatur
- Durchführung / Prüfung

## **Compilerbau / Ziel**

- Verständnis der Funktionsweise eines Compilers
- Verständnis der Mechanismen hinter Programmiersprachen
- Fähigkeit der Entwicklung von Anwendungen mit Analyse formaler Texte
- Implementierung Domänen spezifischer Sprachen
- Grundverständnis im Umgang mit Werkzeugen
- Bytecodes / Laufzeitumgebungen ...

# Compilerbau / Inhalte

- Aufbau eines Compilers
- Theorie (Kontextfreie Sprachen, Endliche Automaten, reguläre Ausdrücke)
- Lexikalische Analyse
- Syntaktische Analyse in LL(1) und LR(1)
- Abstrakter Syntaxbaum, Semantische Analyse
- Codegenierung und Bytecode-Interpreter

## **Compilerbau / Voraussetzung**

- Formal: Hauptstudium
- Kenntnisse der Theoretischen Informatik
- Programmierfähigkeit

# Compilerbau / Werkzeuge

- Java / Eclipse
- JFlex – Scanner Generator (reguläre Ausdrücke → Java)  
<http://jflex.de/>
- Java cup – Parser Generator (Syntaxregeln → Java)  
java cup: <http://www2.cs.tum.edu/projects/cup/>  
CupLex Eclipse-Plugin: <http://cup-lex-eclipse.sourceforge.net/>
- ASM/Bytecodeviewer für Bytecodeverständnis  
<http://asm.ow2.org/>

# Compilerbau / Literatur

- Homepage
- Internet-Materialien
- Sethi, Lam, Aho, Compilerbau, Prinzipien Techniken und Tools  
sehr gut, aber: 70 €  
oder Aho, Sethi, Ullman, Compilerbau, Teil 1 (älter, aber billiger)  
oder aber: Bücher auch aus Bibliothek

## **Compilerbau / Durchführung**

- Zunächst: Theorie, Praxis, Übungen
- Schließt mit Test über Theorie ab.
- Abrundung / Projekte
- Schließt mit Projektvorstellung und Projektabgabe ab