



# Universität Karlsruhe (TH)

## Institut für Innovatives Rechnen und Programmstrukturen (IPD)

Übersetzerbau WS 2003/04

Dozent: Prof. Dr.rer.nat. G. Goos

Übungsleiter: Rubino Geiß

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

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

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

Übungsblatt 9

Ausgabe: 13.02.03

Besprechung: 13.02.03

Auf diesem Blatt stellen wir die Themen der Vorlesung zusammen und geben einige exemplarische Fragen an. Die Fragen sollen weder besondere Relevanz für anstehende Prüfungen signalisieren noch das Fachgebiet vollständig überdecken.

1. **Architektur von Übersetzern:** Definition von Übersetzung, Korrektheit, Struktur, Module, Datenstrukturen

### Analyse

2. **Symbolentschlüsselung** (lexer): Datenstrukturen (Symboltabelle), Reguläre Ausdrücke, Endliche Automaten, Generatoren für Scanner
3. **Zerteilung** (parser):
  - Kontextfreie Grammatik, Konkrete / Abstrakte Syntax
  - LL: Rekursiver Abstieg, Vorschauproblem, Deterministisch-Machen, SLL
  - LR: Kellerklasse, Charakteristischer Automat, Konstruktion, SLR, LALR
  - Fehlerbehandlung: Syntax / Semantik, Systematische Behandlung
4. **Attributierte Grammatiken** (AG): AG Hierarchie (WAG, ANCAG, PAG, OAG, LAG, RAG, AAG), Wohldefiniertheit, Zerlegbarkeit, Geordnete AG, Besuchssequenzen, Abhängigkeiten, Implementierungsansätze
5. **Semantische Analyse:** Namensanalyse, Typanalyse, Operatoridentifikation, Konsistenzprüfung, Sprachabhängige Sonderaufgaben, Strukturäquivalenz, Definitionstabelle, Namenstabelle / Umgebung (Datenstruktur), Überladung vs. Polymorphie

### Abbildung

6. **Transformation:** Zwischendarstellung (Tripel / Quadrupel), Grundblock, Typabbildung, Operatorabbildung, Ablaufsteuerung, Konturmodell, Speicherzuteilung, Parameterübergabereformen
7. **Optimierung:** Wertnummern, CSE, Operatorvereinfachung, Nachoptimierung (ggf. bei Codeerzeugung)
8. **Datenflußanalyse:** vorwärts / rückwärts, möglich / sicher, Intervalle, Registerzuordnung (Färbbarkeit)

### Codierung

9. **Codeerzeugung:** Maschinensimulation, Zielattributierung, Normalformtheorem, Codeauswahl(selektion), Makrosubstitution, Termersetzung (TES, GTES) Codeselektionsgeneratoren (BEG), Globale Registerzuteilung
10. **Assemblierung und Bindung** (assembler & linker): Aufgabe, Speicherabbildung, Bibliotheken, Sprünge

### Exkurs

11. **Speicherbereinigung** (garbage collection): Aufgabenstellung, Relevanz, Ansätze (von Hand, automatisch ohne / mit Informationen), Markierung (Shorr-Waite), Freiliste, mark-and-sweep, mark-and-copy, GC-on-the-fly

## Aufgaben

### 1. Architektur von Übersetzern

- Ihr Compiler läuft zu langsam, woran denken Sie zuerst?
- Der Compiler darf jedes Programm in gewissen Grenzen umgestalten (optimieren), welche sind das?
- Was braucht man für einen Ein-Durchlauf-Übersetzer?
- Wie und wo werden doppelte Vereinbarungen erkannt?
- Was ist eine Hashfunktion? Was bedeutet Kollisionsauflösung?
- Auf welche Module des Compilers greift die Fehlerbehandlung zu?
- Wie würden Sie einen Assembler implementieren?

### 2. Symbolentschlüsselung

- Wie wird ein NEA zu einem äquivalenten DEA? Was heißt hier äquivalent?
- Was liefert die Symbolentschlüsselung an den Zerteiler?
- Wozu braucht man eine Symboltabelle? Wie ist Sie aufgebaut?

### 3. Zerteilung

- Was sind Situationen?
- Kann man eine LL(1) Akzeptor von Hand schreiben? Wenn ja, wie sieht das aus? Geht das vielleicht bei einem LR(1) Akzeptor?
- Wie lautet die SLL(1)-Definition?
- Erklären Sie die Konstruktion des LR(k)-Zerteilers.
- Erklären Sie die Funktionsweise des LR(k)-Zerteilers.
- Wie ist der Zusammenhang zwischen den Kellerklassen und der Konstruktion des LR(k)-Zerteilers.
- Warum ist  $LL(2) \neq SLL(2)$ ?
- Was macht man mit einer Grammatik, die nicht LL(1) ist, man aber nur ein Werkzeug hat, das LL(1) behandeln kann?

### 4. Attributierte Grammatiken

- Was bedeutet LAG(1)?
- Was ist eine zerlegbare AG?
- Wie erfolgt die Reihenfolgebestimmung der Attributberechnungen bei einer OAG?
- Was ist eine OAG?
- Was heißt wohlgeordnet?
- Wie ist ANCAG definiert?
- Wie kommt man zur Zerlegung einer attributierten Grammatik?
- Was ist eine attributierte Grammatik? Welche Klassen kennen Sie?

### 5. Semantische Analyse

- Was liefert der Zerteiler an die semantische Analyse?
- Wie wird der Gültigkeitsbereich von Variablen modelliert?
- Wozu braucht man eine Definitionstabelle?
- Wie wird das Attribut "Umgebung" implementiert?
- Wie führen Sie die Attributierung für die Bezeichnerzuordnung aus?

### 6. Transformation

- Was macht die "Transformation"?
- Was bedeutet Grundblock? Wofür ist er gut?

- (c) Welche Formen der Parameterübergabe kennen Sie? Was bedeutet in diesem Zusammenhang "Kon-turmodell"?

### 7. Optimierung

- (a) Was liefert die Transformation an die Optimierung?
- (b) Wie funktioniert Wertnummerierung?
- (c) Was leistet CSE? Wozu kann man hier Datenflussanalyse brauchen?

### 8. Datenflußanalyse

- (a) Wie betreiben Sie Datenflußanalyse?
- (b) Was heißt eine Analyse ist "sicher" oder "rückwärts"?
- (c) Was heißt Variable  $v$  ist lebendig? Wie berechnet man das?

### 9. Codeerzeugung

- (a) Was ist die Grundidee / technik der Codeerzeugung?
- (b) Wie bestimmt man die Anzahl der benötigten Register bei der Ausdrucksberechnung?
- (c) Welche Registerzuteilungsverfahren kennen Sie?
- (d) Was ist GTES (GTES=Grundtermersetzungssystem)?
- (e) Wie kann man aus einem AST Zielcode machen? Mehrere Möglichkeiten!
- (f) Was sind Register / Wertdeskriptoren?

### 10. Assemblierung und Bindung

- (a) Wie behandeln Sie Sprungmarken?
- (b) Warum gibt es diese Phase in Übersetzern?
- (c) Warum trennt man die Phase durch eine symbolische (Text-) Schnittstelle ab?