

# Optimierende Compiler

LVA 185.A04, VU 2.0, ECTS 3.0

WS 2018/2019

– Vorbesprechung –

(Stand: 01.10.2018)

Jens Knoop



Technische Universität Wien  
Information Systems Engineering  
Compilers and Languages



# Optimierende Übersetzer im WS 2018/19

## Überblick:

- ▶ **LVA:** 185.A04 **Optimierende Übersetzer**, VU 2.0h, 3.0 ECTS  
<http://www.complang.tuwien.ac.at/knoop/oue185A04.html>
- ▶ **Studienrichtung:** Software Engineering & Internet Computing
- ▶ **Ziel:** Vermittlung und Erwerb eines breiten und tiefen Verständnisses für die Grundlagen, Prinzipien und Möglichkeiten von Programmanalysen und -optimierungen in Übersetzern mit dem Ziel, Entwurf und Implementierung von Programmen nicht (mehr) von unnötigen Effizienzüberlegungen überlagern zu lassen.
- ▶ **Vorbesprechung:** Di, 02.10.2018 (15:15-ca.15:45 Uhr), Hörsaal GM7 Kleiner Schiffbau, Getreidemarkt 9.
- ▶ **Vorlesung:** Regelmäßig dienstags 15:15 Uhr - 16:45 Uhr, Hörsaal GM7 Kleiner Schiffbau, Getreidemarkt 9; beginnend Di, 02.10.2018, im Anschluss an die Vorbesprechung.

# Inhaltliche Schwerpunkte

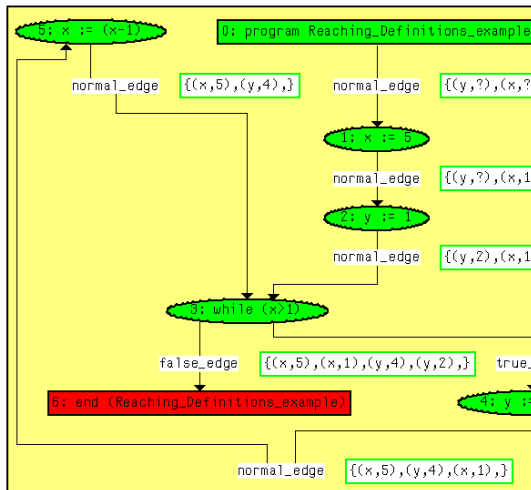
...Übersetzertechnologien für Programmanalyse und -optimierung:

- ▶ Intra- und interprozedurale Datenflussanalyse.
- ▶ Architekturunabhängige Optimierungen.
- ▶ Optimierungen für prozedurale, objektorientierte Sprachen.
- ▶ Termination, Komplexität, Skalierbarkeit.
- ▶ Übungen, i.w. mit Papier und Bleistift.
- ▶ Praktische Übungen, (nach Gegebenheit) mit
  - ▶ SATIrE: Static Analysis Tool Integration Engine (TUW):  
SATIrE = AbsInt-PAG + LLNL-ROSE +...
  - ▶ Spezifikation und Generierung von Programmanalysen.
  - ▶ Analyse von Programmen einer WHILE-Sprache.

# AbsInt-PAG: Programm-Analyse-Generator

...zur Spezifikation, Generierung und Visualisierung von Programmanalysen und deren Ergebnissen:

```
program Reaching_Definitions
[begin]0
[x := 5]1
[y := 1]2
while [(x>1)]3
do (
  [y := (x*y)]4
  [x := (x-1)]5
)
[end]6
```



# Ablauf und Gliederung

- ▶ **Teil I: Einführung**
  - ▶ Motivation, klassische Gen/Kill-Datenflussanalysen.
- ▶ **Teil II: Intraprozedurale Datenflussanalyse (DFA)**
  - ▶ Intraprozeduraler DFA-Rahmen, Konstantenausbreitung und -faltung, partielle Redundanzelimination.
- ▶ **Teil III: Interprozedurale Datenflussanalyse**
  - ▶ Grundlagen, Interprozeduraler DFA-Rahmen, Funktionaler Ansatz, Kontextinformationsansatz, Anwendungen.
- ▶ **Teil IV: Erweiterungen, andere Sprachumgebungen**
  - ▶ Alias- und Heap-Analysen, Optimierungen für objekt-orientierte Programme.
- ▶ **Teil V: Abschluss und Ausblick**

# Literatur

-  Flemming Nielson, Hanne Riis Nielson, Chris Hankin. *Principles of Program Analysis*. Springer-V., 2. Auflage, 452 Seiten, ISBN 3-540-65410-0, 2005.
-  Y. N. Srikant, Priti Shankar. *The Compiler Design Handbook: Optimizations & Machine Code Generation*, CRC Press, 1. Auflage, 928 Seiten, ISBN 084931240X, 2002.
-  Keith D. Cooper, Linda Torczon. *Engineering a Compiler*. Morgan Kaufmann, 801 Seiten, ISBN 155860698X, 2003.
-  Steven S. Muchnick. *Advanced Compiler Design and Implementation*. Morgan Kaufmann, 856 Seiten, ISBN 1558603204, 1997.

...weitere detaillierte Literaturhinweise im Verlauf der Vorlesung und in den Vorlesungsunterlagen.

# Anmeldung und nächste Termine

- ▶ **Anmeldung:** In [TISS](#), in 2er- oder 3er-Gruppen, bis [Fr, den 12.10.2018 \(12:00 Uhr\)](#).
- ▶ **Vorbesprechung:**
  - ▶ [Di, 02.10.2018 \(15:15-ca.15.30 Uhr\)](#), GM7 Kleiner Schiffbau; im Anschluss die erste Vorlesungseinheit.
- ▶ **Vorlesung:**
  - ▶ [Di, 02.10.2018 \(15:15-16:45 Uhr\)](#), GM7 Klein. Schiffbau.
  - ▶ [Di, 09.10.2018 \(15:15-16:45 Uhr\)](#), GM7 Klein. Schiffbau.
- ▶ **Übung:**
  - ▶ Erstes Aufgabenblatt: [Di, 09.10.2018](#), [LVA-Webseite](#).
  - ▶ Abgaben: Jeweils zu Beginn der nächstfolgenden Vorlesungseinheit (d.h. in der Vorlesung).
- ▶ **Beurteilung:** Je zur Hälfte gewichtet die Beurteilung der Übungsaufgaben und einer mdl. Prüfung über Vorlesungs- und Übungsstoff am Ende des Semesters.

Für alle [weiteren Termine](#): Siehe [LVA-Webseite!](#)