

## Vorlesung Compilerbau

Compiler übersetzen Programme aus einer Ausgangssprache in eine Zielsprache, üblicherweise aus einer höheren Programmiersprache in eine Maschinensprache. Dies erlaubt es unter anderem, die gleiche Anwendung mit relativ geringem Aufwand für unterschiedliche Prozessorarchitekturen zu übersetzen und darauf auszuführen. Ein tiefgehendes Verständnis von Compilern hilft dabei, effizientere und sicherere Programme zu schreiben.

Die zu übersetzenden Programme werden außerdem durch Compiler überprüft und verbessert. Wie viele Fehler dabei gefunden werden können, hängt von der Striktheit der Sprache ab. Einige Sprachen, wie z. B. Rust, machen es schwieriger, Programme ohne Fehler zu schreiben, bieten dafür aber gewisse Korrektheitsgarantien. Darüber hinaus führen Compiler eine Vielzahl an Optimierungen durch, um die gegebene Rechnerarchitektur möglichst effizient zu nutzen.

In der Vorlesung werden die Konzepte vermittelt, die Übungen dienen der praktischen Anwendung und Umsetzung der erworbenen Kenntnisse.

- ▶ Vorlesung: Dienstag 11:15-12:45 Uhr (G22A-208)
- ▶ Übungen: Mittwoch 09:15-10:45 Uhr (G29-K059)
- ▶ Kontakt: Michael Kuhn und Michael Blesel

Die Teilnehmenden lernen, wie Compiler funktionieren und welche grundlegenden Konzepte dabei zum Einsatz kommen. In der Übungen werden die vermittelten Kenntnisse praktisch umgesetzt.

Empfohlene Vorkenntnisse:

- ▶ Praktische Kenntnisse in einer Systemprogrammiersprache wie C oder C++
- ▶ Grundkenntnisse in Datenstrukturen und Rechnerarchitekturen

- ▶ 15.10.2024: Introduction (Folien, Materialien)
  - ▶ 22.10.2024: Scanning (Folien)
  - ▶ 29.10.2024: Parsing (Part 1) (Folien)
  - ▶ 05.11.2024: Parsing (Part 2) (Folien)
  - ▶ 12.11.2024: Abstract Syntax Trees (Folien)
  - ▶ 19.11.2024: Semantic Analysis (Folien)
  - ▶ 26.11.2024: Intermediate Representation (Folien)
  - ▶ 03.12.2024: Memory Organization (Folien)
  - ▶ 10.12.2024: Assembly Language (Folien)
  - ▶ 17.12.2024: Code Generation (Folien)
  - ▶ 07.01.2025: Fällt aus
  - ▶ 14.01.2025: Optimization (Folien)
  - ▶ 21.01.2025: Research Talk
  - ▶ 28.01.2025: Nachbesprechung
- 
- ▶ 15.10.2024: Einführung (Blatt 0, Blatt 1, Materialien)
    - ▶ Deadline: 27.10.2024, 23:59
  - ▶ 28.10.2024: Scanning (Blatt 2, Materialien)
    - ▶ Deadline: 10.11.2024, 23:59
  - ▶ 12.11.2024: Parsing (Blatt 3, Materialien)
    - ▶ Deadline: 01.12.2024, 23:59

- ▶ 03.12.2024: AST (Blatt 4, Materialien)
    - ▶ Deadline: 12.01.2025, 23:59
  - ▶ 14.01.2025: Semantic Analysis (Blatt 5, Materialien)
    - ▶ Deadline: 26.01.2025, 23:59
  - ▶ TBD: TBD (Blatt 6, Materialien)
    - ▶ Deadline: TBD, 23:59
  - ▶ TBD: TBD (Blatt 7, Materialien)
    - ▶ Deadline: TBD, 23:59
  - ▶ TBD: TBD (Blatt 8, Materialien)
    - ▶ Deadline: TBD, 23:59
- 
- ▶ Introduction to Compilers and Language Design (Douglas Thain)
-