

| | | | |
|----------------------------|--|------------------------|---------------------------------------|
| Lehrende/r | Prof. Dr. Friedrich Steimann | Modulverantwortliche/r | Prof. Dr. Friedrich Steimann |
| | Dauer des Moduls ein Semester | ECTS 10 | Workload 300 Stunden |
| | | | Häufigkeit in jedem Wintersemester |
| Lehrveranstaltung(en) | Übersetzerbau | | |
| Detaillierter Zeitaufwand | Bearbeiten der Lektionen: 150 Stunden Bearbeitung der Einsendeaufgaben: 75 Stunden Wiederholung und Prüfungsvorbereitung, Prüfung: 75 Stunden | | |
| Qualifikationsziele | Nach erfolgreicher Teilnahme besitzen Studierende einen guten Überblick über die Techniken des Übersetzerbaus. Sie können erklären, mit welchen Schritten höhere Programmiersprachen in Maschinensprache überführt werden. Sie sind in der Lage, eigene kleine Programmiersprachen für spezielle Anwendungen, komplexe Dateiformate oder Protokolle für die Client-Server-Kommunikation zu definieren und dafür mit Hilfe von Werkzeugen wie Lex/Yacc (Flex/Bison) entsprechende lexikalische Analysatoren und Parser zu konstruieren. Sie können Definitionen für wesentliche Begriffe der zugrundeliegenden Theorie angeben, etwa für kontextfreie Grammatiken, reguläre Sprachen/Ausdrücke, endliche Automaten, attributierte Grammatiken. Sie können die Konstruktion und Arbeitsweise von Top-Down- oder Bottom-Up-Parsern im Detail erklären. | | |
| Inhalte | <p>Das Modul behandelt Techniken und Werkzeuge zur Übersetzung einer (formalen) Quellsprache in eine Zielsprache. Zumindest die Quellsprache ist durch eine geeignete Grammatik definiert. Der klassische Anwendungsfall ist die Übersetzung von einer höheren Programmiersprache in Maschinen- oder Assemblersprache. Übersetzungstechniken haben aber viel weitergehende Anwendbarkeit: Andere Quellsprachen können z.B. Dokumentbeschreibungssprachen sein (LaTeX, HTML), Anfragesprachen in Datenbanksystemen, VLSI-Entwurfssprachen usw. oder auch komplex strukturierte Eingabedateien für Anwendungsprogramme. Es gibt mächtige Werkzeuge (Scanner- und Parsergeneratoren auf der Basis attributierter Grammatiken), deren Kenntnis für jeden Informatiker nützlich ist, auch wenn nur wenige klassische Übersetzer (Compiler) implementieren.</p> <p>Lexikalische Analyse (Scanner), Syntaxanalyse (Parser), Syntaxgesteuerte Übersetzung, Übersetzung einer Dokument-Beschreibungssprache, Übersetzung imperativer Programmiersprachen, Übersetzung funktionaler Programmiersprachen, Codeerzeugung und Optimierung.</p> <p>In den Übungen wird u.a. ein größeres durchgängiges Projekt zum Compilerbau bearbeitet.</p> <p>Ergänzende Literatur: A.V. Aho, M.S. Lam, R. Sethi, J.D. Ullman: Compilers: Principles, Techniques, and Tools. 2. Aufl. Addison-Wesley, 2006. R. Wilhelm, D. Maurer, Übersetzerbau: Theorie, Konstruktion, Generierung. 2. Aufl., Springer-Verlag, 2007. R.H. Güting, M. Erwig: Übersetzerbau: Techniken, Werkzeuge, Anwendungen. Springer-Verlag, 1999.</p> | | |
| Inhaltliche Voraussetzung | Grundbegriffe der Theorie der formalen Sprachen sind nützlich, werden aber auch in dieser Lehrveranstaltung vermittelt. | | |
| Lehr- und Betreuungsformen | Lehrveranstaltungsmaterial Einsendeaufgaben mit Korrektur und/oder Musterlösung internetgestütztes Diskussionsforum | | |

| | | | |
|-------------------------|------|---|----------------------|
| | | Betreuung und Beratung durch Lehrende | |
| Anmerkung | | - | |
| Formale Voraussetzung | | Wahlmodul I: mindestens 30 von 60 ECTS der Studieneingangsphase sind bestanden; Wahlmodule II-IV: Studieneingangsphase ist abgeschlossen, die Module Grundpraktikum Programmierung, Grundlagen der Theoretischen Informatik und Softwaresysteme sind bestanden | |
| Verwendung des Moduls | | B.Sc. Informatik B.Sc. Mathematisch-technische Softwareentwicklung B.Sc. Wirtschaftsinformatik M.Sc. Informatik M.Sc. Praktische Informatik M.Sc. Wirtschaftsinformatik | |
| Prüfungsformen | | Art der Prüfungsleistung | Voraussetzung |
| Prüfung | | benotete mündliche Prüfung (ca. 25 | keine |
| Stellenwert der Note | 1/16 | Minuten) | |