

<b>Übersetzerbau</b> <i>Compiler Construction</i>							Modulnummer:		
Bachelor Pflicht <input type="checkbox"/> Winf-Schwerpunkt-Pflicht <input type="checkbox"/> Winf-Schwerpunkt-Wahlpflicht <input type="checkbox"/> Winf-Wahl <input type="checkbox"/>				Schwerpunkt Computational Finance <input type="checkbox"/> E-Business <input type="checkbox"/> IT-Management <input type="checkbox"/> Logistik <input type="checkbox"/>					
Anzahl der SWS	V 3	UE 1	K 0	S 0	Prak. 0	Proj. 0	$\Sigma$ 4	Kreditpunkte: 6	Turnus i. d. R. angeboten alle 2 Semester
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: Theoretische Informatik 1, Theoretische Informatik 2									
Vorgesehenes Semester: ab 5. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> <li>• Prinzipien der Strukturierung von Übersetzern und Interpretern verstehen und anwenden können</li> <li>• Konzepte und Methoden der lexikalischen, syntaktischen und kontextuellen (statisch semantischen) Analyse verstehen, anwenden, auf die Implementierung konkreter Sprachen übertragen, beurteilen und bewerten können</li> <li>• Prinzipien der Übersetzung von imperativen und objektorientierten Programmiersprachen in Maschinencode verstehen, auf die Implementierung konkreter Konzepte übertragen und die Qualität des Codes beurteilen können.</li> <li>• Prinzipien der Codeerzeugung (Registerzuteilung, Instruktionsauswahl, globale und lokale Optimierung) verstehen können</li> <li>• selbstständig und in kleinen Teams Wissen und Verständnis erwerben und darstellen können.</li> </ul>									
Inhalte: <ul style="list-style-type: none"> <li>• Implementierung von Programmiersprachen mit Interpretern, und Übersetzern.</li> <li>• Strukturierung von Übersetzern: Plattform(un)abhängigkeit, Bootstrap, Phasen.</li> <li>• Lexikalische Analyse: reguläre Definitionen, endliche Automaten, Symboltabellen, Benutzung von flex.</li> <li>• Syntaxanalyse: kontextfreie Grammatiken, ab- und aufsteigendes Parsieren, Baumaufbau, Fehlerbehandlung, Benutzung von bison.</li> <li>• Kontext-Analyse: Attributgrammatiken, Auswerter, Vereinbarungstabellen.</li> <li>• Transformation von imperativen und objektorientierten Programmen in abstrakten Maschinencode.</li> <li>• Grundzüge der Codeerzeugung für konkrete Maschinen: globale Optimierung, Registerzuteilung, Instruktionsauswahl, lokale Optimierung.</li> </ul> <p>In der Übung Anwendung der in der Vorlesung erworbenen Kenntnisse und Fähigkeiten auf spezifische Konstrukte von Programmiersprachen.</p> <p>Insbesondere werden folgende theoretisch/methodische Grundlagen behandelt:</p> <ul style="list-style-type: none"> <li>• Theorie der regulären und kontextfreien Sprachen</li> <li>• Algorithmen zur Konstruktion von deterministischen endlichen Automaten für reguläre Definitionen</li> <li>• Theorie des LL(k) und LR(k)-Parsierens, mit automatischer Fehlerbehandlung</li> <li>• Methoden der Grammatikdefinition, -transformation und -disambiguierung.</li> <li>• Theorie der Zweistufengrammatiken und Attributgrammatiken</li> <li>• Algorithmen zum Erzeugens von Auswertern für Attributgrammatiken</li> <li>• Methoden der Spezifikation von abstrakten Datentypen, für Bezeichnertabellen und Vereinbarungstabellen</li> <li>• Methodik der rekursiven Syntax-orientierten Definition für die Transformation von Syntaxbäumen in abstrakten Maschinencode</li> </ul>									

Unterlagen (Skripte, Literatur, Programme usw.):

- A.V. Aho, M. S. Lam, R. Sethi, J.D. Ullman. Compilers - Prinzipien, Techniken und Werkzeuge, zweite Auflage, Bonn: Pearson Education Deutschland (2008).
- R. Wilhelm, D. Maurer. Übersetzerbau: Theorie - Konstruktion - Generierung. Berlin: Springer, 2. Auflage (1997).

Weiteres Lehrmaterial ist auf der Webseite der Veranstaltung zu finden:

- Folienkopien
- Übungsaufgaben.

Übersetzer-Werkzeuge lex/flex, yacc/bison stehen im Rechnernetz des Studiengangs zur Verfügung.

Form der Prüfung:  
i.d.R. mündliche Prüfung

Arbeitsaufwand	Präsenz	56 h
	Übungsbetrieb/Prüfungsvorbereitung	124 h
	Summe	180 h

Lehrende:  
Dr. B. Hoffmann

Verantwortlich:  
Dr. B. Hoffmann