

Modulbezeichnung	Übersetzer-Praktikum								
Modulverantwortliche(r)	Dr. B. Hoffmann								
Modulart	Pflicht/Wahl <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>								
Spezialisierungsbereich									
Dauer des Moduls	1 Semester								
Kreditpunkte	4 CP								
Arbeitsaufwand	<table style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2">Berechnung des Workloads</td> </tr> <tr> <td style="width: 80%;">Präsenz</td> <td style="text-align: right;">28 h</td> </tr> <tr> <td>Übungsbetrieb/Prüfungsvorbereitung</td> <td style="text-align: right;">92 h</td> </tr> <tr> <td style="border-top: 1px solid black;">Summe</td> <td style="text-align: right; border-top: 1px solid black;">120 h</td> </tr> </table>	Berechnung des Workloads		Präsenz	28 h	Übungsbetrieb/Prüfungsvorbereitung	92 h	Summe	120 h
Berechnung des Workloads									
Präsenz	28 h								
Übungsbetrieb/Prüfungsvorbereitung	92 h								
Summe	120 h								
Turnus des Moduls	i. d. R. angeboten alle 2 Jahre								
Voraussetzung für die Teilnahme	Keine <input type="checkbox"/> Folgende Inhaltliche Voraussetzungen: Übersetzer								
Lehr- und Lernformen	Seminar <input type="checkbox"/> Vorlesung <input checked="" type="checkbox"/> Tutorium <input checked="" type="checkbox"/> Praktikum <input checked="" type="checkbox"/> Projekt <input type="checkbox"/>								
Lernziele	<ul style="list-style-type: none"> • Konzepte, Methoden und Algorithmen zur Implementierung imperativer und objektorientierter Programmiersprachen auf eine konkrete Programmiersprache übertragen können • Größere Programme nach den Regeln der Softwaretechnik in kleinen Teams verstehen, erweitern und vertiefen können • Über Erfahrung in der projektbezogenen Problemlösung in kleinen Teams verfügen. 								
Lerninhalte	<ul style="list-style-type: none"> • Implementierung der lexikalischen Analyse (Transformation regulärer Definitionen in endliche Automaten, Implementierung von Symboltabellen). • Implementierung der Syntaxanalyse (Transformation von kontextfreien Grammatiken in absteigende Parsierer, Implementierung der Fehlerbehandlung und des abstrakten Syntaxbaums). • Implementierung der Kontext-Analyse (Entwicklung rekursiver Baumauswerter, Implementierung von Vereinbarungstabellen). • Erzeugung von abstraktem Maschinencode für eine objektorientierte Programmiersprache. 								
Prüfungsformen	Semesterarbeit (Implementierung eines Übersetzers)								
Literatur	<ul style="list-style-type: none"> • A. W. Appel: Modern Compiler Implementation in Java, Cambridge University Press 1998. • D. A. Watt und D. F. Brown: Programming Language Processors in Java - Compilers and Interpreters, Prentice-Hall, 2000. <p>Weiteres Lehrmaterial ist auf der Webseite des Veranstaltung zu finden:</p> <ul style="list-style-type: none"> • Entwicklungsumgebung für die Implementierung von Übersetzern (Oberon-System) • Ein Übersetzer und Interpreter (das PL0-System) • Aufgabenbeschreibung 								