

Programmiersprachen <i>Programming Languages</i>							Modulnummer: MB-705.01													
Master Pflicht/Wahl <input type="checkbox"/> Wahl <input checked="" type="checkbox"/> Basis <input checked="" type="checkbox"/> Ergänzung <input type="checkbox"/> Sonderfall <input type="checkbox"/>				Zugeordnet zu Masterprofil <table style="width: 100%; border: none;"> <tr> <td></td> <td style="text-align: right;">Basis</td> <td style="text-align: right;">Ergänzung</td> </tr> <tr> <td>Sicherheit und Qualität (SQ)</td> <td style="text-align: right;"><input checked="" type="checkbox"/></td> <td style="text-align: right;"><input type="checkbox"/></td> </tr> <tr> <td>KI, Kognition, Robotik (KIKR)</td> <td style="text-align: right;"><input type="checkbox"/></td> <td style="text-align: right;"><input type="checkbox"/></td> </tr> <tr> <td>Digitale Medien und Interaktion (DMI)</td> <td style="text-align: right;"><input type="checkbox"/></td> <td style="text-align: right;"><input type="checkbox"/></td> </tr> </table>						Basis	Ergänzung	Sicherheit und Qualität (SQ)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	KI, Kognition, Robotik (KIKR)	<input type="checkbox"/>	<input type="checkbox"/>	Digitale Medien und Interaktion (DMI)	<input type="checkbox"/>	<input type="checkbox"/>
	Basis	Ergänzung																		
Sicherheit und Qualität (SQ)	<input checked="" type="checkbox"/>	<input type="checkbox"/>																		
KI, Kognition, Robotik (KIKR)	<input type="checkbox"/>	<input type="checkbox"/>																		
Digitale Medien und Interaktion (DMI)	<input type="checkbox"/>	<input type="checkbox"/>																		
Modulbereich: Praktische und Technische Informatik Modulteilbereich: 705 Programmiersprachen und Übersetzer																				
Anzahl der SWS		<table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;">V</td> <td style="padding: 2px;">UE</td> <td style="padding: 2px;">K</td> <td style="padding: 2px;">S</td> <td style="padding: 2px;">Prak.</td> <td style="padding: 2px;">Proj.</td> <td style="padding: 2px;">Σ</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">4</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">4</td> </tr> </table>	V	UE	K	S	Prak.	Proj.	Σ	0	0	4	0	0	0	4	Kreditpunkte: 6		Turnus i. d. R. angeboten alle 2 Jahre	
V	UE	K	S	Prak.	Proj.	Σ														
0	0	4	0	0	0	4														
Formale Voraussetzungen: -																				
Inhaltliche Voraussetzungen: Praktische Informatik 1, Praktische Informatik 2, Praktische Informatik 3																				
Vorgesehenes Semester: ab 1. Semester																				
Sprache: Deutsch																				
Ziele: <ul style="list-style-type: none"> ● Beschreibungen von Programmiersprachen verstehen und in Hinblick auf Konzepte, auf die Unterstützung von Programmier-Paradigmen und auf Entwurfsziele analysieren können ● Ausprägungen von Konzepten und Paradigmen in verschiedenen Programmiersprache vergleichen und bewerten können ● hinterfragen, wie weit Programmiersprachen ein Programmierparadigma unterstützen und die von ihren Entwerfern gesteckten Entwurfsziele erreichen 																				
Inhalte: Konzepte <ul style="list-style-type: none"> ● Werte (Datenstrukturen und Ausdrücke). ● Speicher (Variablen und Befehle) ● Bindung (Vereinbarungen und Gültigkeitsbereiche). ● Abstraktion (Funktionen, Prozeduren und Parameterübergabe). ● Kapselung (Moduln, abstrakte Datentypen, Klassen, generische Pakete). ● Typsysteme (Überladen, Anpassungen, Polymorphie, Untertypen und Vererbung). ● Ablaufsteuerung (Sprünge, Ausweg, Ausnahmen). ● Nebenläufigkeit und Verteiltheit Paradigmen (Programmierstile) <ul style="list-style-type: none"> ● Imperatives Programmieren. ● Objekt-orientiertes Programmieren. ● Nebenläufiges Programmieren. ● Funktionales Programmieren. ● Logisches Programmieren. Prinzipien des Sprachentwurfs <ul style="list-style-type: none"> ● Syntax. ● Semantik. ● Pragmatik. <p>In der Übung Anwendung der in der Vorlesung erworbenen Kenntnisse und Fähigkeiten bei der Untersuchung spezifischer Konzepte und Eigenschaften von spezifischer Programmiersprachen (z. B. Ada, Eiffel, Java, Haskell, Prolog)</p>																				

Unterlagen (Skripte, Literatur, Programme usw.):

- David A. Watt: Programming Language Design Concepts, Chichester: Wiley and Sons (2004).
- Robert W. Sebesta: Concepts of Programming Languages 5/e, Reading, MA: Addison-Wesley (2002).

Weiteres Lehrmaterial ist auf der Webseite der Veranstaltung zu finden:

- Online-Fassung des Buches David A. Watt: Programmiersprachen - Konzepte und Paradigmen, München-Wien: Hanser (1996)
- Folienkopien
- Übungsaufgaben
- Beschreibungen der Referenzsprachen Ada, Eiffel, Java, Haskell, Prolog
- Hinweise auf Quellen im WWW

Implementierungen der Referenzsprachen Ada, Eiffel, Java, Haskell, Prolog 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