

Modul entfällt zum Wintersemester 2019/20

BOSS-NR. 87300

Modul INF-BSc-312: Einführung in den logisch-algebraischen Systementwurf (ELAS)					
Englischer Modultitel: Introduction to Logic-Algebraic System Design					
Studiengänge: Bachelorstudiengang Informatik , Bachelorstudiengang Angewandte Informatik					
Turnus nach Ankündigung	Dauer 1 Semester	Studienabschnitt 5.-6. Semester	Credits 4	Aufwand 120 (45/75)	
1	Modulstruktur				
	Nr.	Element / Lehrveranstaltung	Typ	Credits	SWS
	1	Logisch-algebraischer Systementwurf (Einführung)	V	2	2
	2	Übungen zu logisch-algebraischer Systementwurf (Einführung)	Ü	2	1
2	Lehrveranstaltungssprache: deutsch				
3	<p>Lehrinhalte</p> <p>Die Lehrveranstaltung behandelt die grundlegenden Konzepte zu Konstruktion, Ausführung und Verifikation formaler Modelle. 40 Jahre Forschung und Entwicklung an der Schnittstelle zwischen Mathematik und Softwaretechnik haben zu der heutigen Sicht geführt, nach der jedes Modell entweder konstruktor- oder destruktorbasiert ist, entweder über den Aufbau oder das Verhalten seiner Objekte definiert ist. Zur ersten Gruppe von Modellen gehören all diejenigen, deren Elemente aus endlich vielen Komponenten bestehen und die oft durch Grammatiken beschrieben werden. Die zweite Gruppe umfasst Automaten, Kripke-Strukturen, Petri-Netze, Ströme, Term- oder Flussgraphen, Prozessalgebren, Klassenhierarchien, kurzum alle durch Zustandsübergänge und/oder Attribute charakterisierten Strukturen. Theoretische Untersuchungen ebenso wie praktische Anwendungen haben gezeigt, dass beide Modellklassen dual zueinander sind und diese Dualität von der Datenstrukturierung bis zu den Rechen- und Beweismethoden reicht.</p> <p>Im einzelnen werden nach einer Wiederholung der mengentheoretischen Strukturierungskonzepte Produkt, Summe, Quotient und Teilmenge – u. a. diese Konzepte verallgemeinernde – Grundbegriffe der Kategorientheorie eingeführt: Kategorien, Funktoren, natürliche Transformationen, Limiten, Colimiten, Algebren, Coalgebren, freie und cofreie Strukturen. In den Kategorien mehrsortiger Mengen bzw. CPOs (Mengen mit kettenvollständiger Halbordnung) bilden diese Konstruktionen die semantische Grundlage für eine universelle Spezifikationsprache, die rein funktionale ebenso wie prädikaten-, modal- und temporallogische Ansätze umfasst. Die Sprache dient nicht nur der Beschreibung des jeweils modellierten Systems, sondern bestimmt auch erstens dessen Ausführung im Sinne der Auswertung von Ausdrücken und der Lösung von Gleichungen, zweitens die Verifikation von Anforderungen an das System und drittens seine Optimierung bezüglich unterschiedlicher Zielfunktionen. Ausführung, Verifikation und Optimierung basieren auf Regelsystemen und Strategien ihrer Anwendung, die neben den o. g. semantischen Konzepten den Hauptinhalt der Lehrveranstaltung bilden.</p> <p>Als Rechnerunterstützung und Illustrationsmittel wird von Anfang an das in einer objektorientierten Erweiterung von Haskell codierte Entwurfs- und Beweiswerkzeug Expander2 eingesetzt. Damit lässt sich u. a. eine Brücke schlagen von der Vermittlung theoretischer Ansätze zu Anwendungs- oder Implementierungsproblemen und deren Lösungen, was umgekehrt entscheidende Kriterien für Auswahl, Anpassung und Weiterentwicklung der theoretischen Ansätze liefert.</p>				
4	<p>Kompetenzen</p> <p>Die Studierenden lernen einzuschätzen, welche mathematischen Konstruktionen und darauf aufbauenden Werkzeuge für welche Anwendungen geeignet bzw. nicht geeignet sind und wie man sie an spezielle Anforderungen anpassen kann. Eignung entsteht nicht nur durch eine adäquate, präzise Syntax und Semantik, sondern auch durch den Einsatz von Methoden, die nachvollziehbares und möglichst effizientes Rechnen, Lösen und Beweisen in den jeweiligen Modellen ermöglichen. Da Logik und Algebra nicht nur den klarsten begrifflichen Rahmen, sondern auch die mächtigsten und flexibelsten Verfahren bieten, um Softwarespezifikationen zu erstellen und zu analysieren, ist der hinreichend souveräne Umgang mit logisch-algebraischen Techniken das vordringliche Lernziel.</p>				

5	Prüfungen <i>Modulprüfung:</i> mündliche Prüfung ^{BOSS-NR. 87391} <i>Studienleistung:</i> <ul style="list-style-type: none"> • Aktive Mitarbeit in den Übungen; Erreichen der Mindestpunktezahl der Übungsaufgaben ^{BOSS-NR. 87341} Die Studienleistung ist Voraussetzung für die Teilnahme an der Modulprüfung.			
6	Prüfungsformen und -leistungen <input checked="" type="checkbox"/> Modulprüfung <input type="checkbox"/> Teilleistungen			
7	Teilnahmevoraussetzungen <i>Erfolgreich abgeschlossen:</i> –keine– <i>Vorausgesetzte Kenntnisse:</i> Modul „Datenstrukturen, Algorithmen und Programmierung 1 (DAP1)“, Modul „Funktionale Programmierung,“ Modul „Logik für Informatik“ <i>Wünschenswerte Kenntnisse:</i> Modul „Software-Technik (SWT)“ ¹ , Modul „Grundbegriffe der theoretischen Informatik (GTI)“			
8	Modultyp und Verwendbarkeit des Moduls Wahlmodul in den Bachelor-Studiengängen Informatik und Angewandte Informatik			
9	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; padding: 5px;"> Modulbeauftragte/r Prof. Dr. P. Padawitz </td> <td style="width: 30%; padding: 5px;"> Zuständige Fakultät Informatik </td> <td style="width: 20%; padding: 5px; font-size: small;"> Beschluss Fakultätsrat 03.06.2009 Änderung Fakultätsrat 21.05.2014 Außerkraftsetzung Fakultätsrat 23.10.2010 </td> </tr> </table>	Modulbeauftragte/r Prof. Dr. P. Padawitz	Zuständige Fakultät Informatik	Beschluss Fakultätsrat 03.06.2009 Änderung Fakultätsrat 21.05.2014 Außerkraftsetzung Fakultätsrat 23.10.2010
Modulbeauftragte/r Prof. Dr. P. Padawitz	Zuständige Fakultät Informatik	Beschluss Fakultätsrat 03.06.2009 Änderung Fakultätsrat 21.05.2014 Außerkraftsetzung Fakultätsrat 23.10.2010		

¹ oder außer Kraft gesetztes Modul „Software-Entwicklung (SE)“