

<b>Modul INF-BSc-254: Fachprojekt „Rapid Prototyping mit Expander 2/3“</b>					
<b>Englischer Modultitel:</b> Undergraduate Project „Rapid Prototyping with Expander 2/3“					
<b>Studiengänge:</b> Bachelorstudiengang Informatik , Bachelorstudiengang Angewandte Informatik					
<b>Turnus</b> nach Ankündigung	<b>Dauer</b> 1 Semester	<b>Studienabschnitt</b> 6. Semester		<b>Credits</b> 7 <sup>1</sup>	<b>Aufwand</b> 210 (60/150)
<b>1</b>	<b>Modulstruktur</b>				
	<b>Nr.</b>	<b>Element / Lehrveranstaltung</b>	<b>Typ</b>	<b>Credits</b>	<b>SWS</b>
	1.	Fachprojekt: „Rapid Prototyping mit Expander 2/3“	Projekt	7	4
<b>2</b>	<b>Lehrveranstaltungssprache:</b> deutsch				
<b>3</b>	<p><b>Lehrinhalte</b></p> <p>Expander2/3 ist ein Haskell-basiertes Spezifikations-, Verifikations- und Darstellungswerkzeug mit einer Tcl/Tk- bzw. GTK-Schnittstelle für graphische Darstellungen, Interaktionen und Animationen. Den Prinzipien funktionaler Programmierung folgend, sind die Features von Expander2/3 orthogonal organisiert und daher offen für Spezialisierungen und Erweiterungen, sowohl der (als Haskell-Programme) integrierten Algorithmen als auch von BenutzerInnen spezifizierter formaler Modelle, die mit dem System ausgeführt werden können.</p> <p>Die Projektaufgabe wird zu Beginn der LV in Abstimmung mit ihren TeilnehmerInnen und deren jeweiligen Kenntnissen und Studienschwerpunkten festgelegt. Sie kann eine Spezialisierung oder Erweiterung in obigem Sinne sein oder in der Entwicklung von Spezifikationen in einem bestimmten Anwendungsbereich bestehen.</p> <p>Die Spezifikationsprache von Expander2/3 umfasst Funktionen und Prädikate höherer Ordnung, die Anfrage-, Beschreibungs- oder dynamische Logiken charakterisieren. Insbesondere können beliebige Transitionssysteme mit strukturierten Zuständen regelbasiert spezifiziert und durch direkte Auswertung von Anfragen (Model Checking) oder mit Hilfe allgemeinerer Beweisregeln wie (Co-)Resolution und (Co-)Induktion verifiziert werden. Letztere erfordern Interaktion, können aber auch auf parametrisierte Transitionssysteme angewendet werden, während Model Checking im Sinne der Auswertung logischer Formeln automatisch ablaufen kann - bzgl. konkreter Modelle, die auch außerhalb von Expander2/3 erzeugt und dann an das System zur Weiterverarbeitung übergeben werden können.</p>				
<b>4</b>	<p><b>Kompetenzen</b></p> <p>Die Studierenden lernen, komplexe Aufgabenstellungen logisch-algebraisch zu formulieren und Lösungen in einer darauf zugeschnittenen Programmierumgebung zu testen und schrittweise zu optimieren. Hierzu müssen Inhalte verschiedener LVs zusammengebracht werden sowie Logik und funktionale Programmierung kennzeichnende Konzepte in den abstrakten Entwurf und die konkrete Implementierung einfließen.</p>				
<b>5</b>	<p><b>Prüfungen</b></p> <p><i>Voraussetzung für den Modulabschluss:</i><sup>2</sup></p> <ul style="list-style-type: none"> <li>• mündliche Prüfung (30 Minuten) <small>BOSS-NR. 66491</small></li> </ul>				
<b>6</b>	<p><b>Prüfungsformen und -leistungen</b></p> <p><input checked="" type="checkbox"/> Modulprüfung <input type="checkbox"/> Teilleistungen</p>				
<b>7</b>	<p><b>Teilnahmevoraussetzungen</b></p> <p><i>Erfolgreich abgeschlossen:</i> –keine–</p> <p><i>Wünschenswerte Kenntnisse:</i> Funktionale und/oder logische Programmierung, mathematische Grundbegriffe</p>				
<b>8</b>	<p><b>Modultyp und Verwendbarkeit des Moduls</b></p> <p>Wahlpflicht-Modul im Bachelor-Studiengang Informatik und Angewandte Informatik, Fachprojekt</p>				

<sup>1</sup> 6 Leistungspunkte vor dem Wintersemester 2019/20

<sup>2</sup> vor dem Wintersemester unbenotete Modulprüfung

<b>9</b>	<b>Modulbeauftragte/r</b> Prof. Dr. P. Padawitz	<b>Zuständige Fakultät</b> Informatik	Beschluss Fakultätsrat 16.01.2008 Änderung Fakultätsrat 18.01.2012, 21.05.2014, 22.05.2019, 18.10.2022
----------	--	--	--