

Testautomatisierung								Modulnummer:	
<i>Test Automation</i>									
Bachelor Pflicht/Wahl <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/> Wahl <input type="checkbox"/> Sonderfall <input type="checkbox"/>				Modulbereich: Pflicht					
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus i. d. R. angeboten alle 2 Jahre
	0	0	4	0	0	0	4		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: Grundlagen von Test und Verifikation									
Vorgesehenes Semester: ab 1. Semester									
Sprache: Deutsch/Englisch									
Ziele: Die Studierenden verfügen über ein vertieftes Verständnis für <ul style="list-style-type: none"> • Testfallentwurf • Bezug zwischen Anforderungen und Testfällen • Modell-basierte Testfallerzeugung • Algorithmen für die automatische Testfall-/Testdatenerzeugung • Äquivalenz zwischen erschöpfenden Tests und Korrektheitsbeweis 									
Inhalte: <ol style="list-style-type: none"> 1. Vorgehensmodelle und Testprozess 2. Testarten auf unterschiedlichen Systemebenen 3. Modell-basiertes Testen - die W-Methode von Chow 4. Strukturelles Testen 5. Modell-basiertes Testen von Echtzeitsystemen 6. Spezialthemen aus den Gebieten <ul style="list-style-type: none"> • SMT-Solver für die Berechnung konkreter Testdaten • Äquivalenzklassentests für nebenläufige Echtzeitsysteme • Überdeckungskriterien und ihr Bezug zum Korrektheitsbeweis • Mutationstests 									
Unterlagen (Skripte, Literatur, Programme usw.): <ul style="list-style-type: none"> • R. Binder "Testing Object-Oriented Systems: Models, Patterns, and Tools", Addison-Wesley, 2000 • A. Spillner, T. Linz "Basiswissen Softwaretest: Aus- und Weiterbildung zum Certified-Tester", dpunkt-Verlag, 2003. • J. Peleska und M. Siegel "Test Automation of Safety-Critical Reactive Systems", South African Computer Journal, No. 19, pp. 53-77, 1997. • J. Peleska "Formal Methods and the Development of Dependable Systems", Habilitationsschrift, Bericht Nr. 9612, Dezember 1996, Institut für Informatik und praktische Mathematik, Christian-Albrechts-Universität Kiel, 1997. • Tsun S. Chow "Testing Software Design Modeled by Finite-State Machines", IEEE Transactions on Software Engineering, SE-4(3), pp. 178-186, März 1978. 									
Form der Prüfung: i.d.R. Bearbeitung von Übungsaufgaben und Fachgespräch oder mündliche Prüfung									
Arbeitsaufwand		Präsenz		56 h		Übungsbetrieb/Prüfungsvorbereitung		124 h	
		Summe		180 h					

Lehrende:
Prof. Dr. J. Peleska

Verantwortlich:
Prof. Dr. J. Peleska