

Modulbezeichnung	Grundlagen von Test und Verifikation (deleted:Tue Jul 26 10:39:26 +0200 2011)								
Modulverantwortliche(r)	Prof. Dr. J. Peleska								
Modulart	Pflicht/Wahl <input type="checkbox"/> Wahlpflicht <input checked="" type="checkbox"/>								
Spezialisierungsbereich	Systemsoftware / Eingebettete Systeme								
Dauer des Moduls	1 Semester								
Kreditpunkte	6 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;">56 h</td> </tr> <tr> <td>Übungsbetrieb/Prüfungsvorbereitung</td> <td style="text-align: right;">124 h</td> </tr> <tr> <td>Summe</td> <td style="text-align: right; border-top: 1px solid black;">180 h</td> </tr> </table>	Berechnung des Workloads		Präsenz	56 h	Übungsbetrieb/Prüfungsvorbereitung	124 h	Summe	180 h
Berechnung des Workloads									
Präsenz	56 h								
Übungsbetrieb/Prüfungsvorbereitung	124 h								
Summe	180 h								
Turnus des Moduls	i. d. R. angeboten in jedem WiSe								
Voraussetzung für die Teilnahme	Keine <input type="checkbox"/> Folgende Inhaltliche Voraussetzungen: Technische Informatik 2								
Lehr- und Lernformen	Seminar <input type="checkbox"/> Vorlesung <input checked="" type="checkbox"/> Tutorium <input checked="" type="checkbox"/> Praktikum <input type="checkbox"/> Projekt <input type="checkbox"/>								
Lernziele	<ul style="list-style-type: none"> • Testfälle entwerfen und in Bezug zu Anforderungen setzen können • Testüberdeckungs- und Testendekriterien beurteilen können • Korrektheitsbeweise auf Basis der Programmsemantik durchführen • Schaltkreisverifikation durch Eigenschaftsprüfung verstehen • Den Zusammenhang zwischen Modelläquivalenz und erschöpfendem Testen, sowie formalem Äquivalenzbeweis verstehen • Gemeinsamkeiten und Unterschiede bei Test und Verifikation von Software und Schaltkreisen verstehen 								
Lerninhalte	<ol style="list-style-type: none"> 1. Der Testprozess in den Vorgehensmodellen des Software-Engineering 2. Testen auf unterschiedlichen Integrationsebenen: Vom Modultest bis zum Systemtest 3. Test-Abdeckungsmaße für Softwaretests 4. Fehlermodelle und -überdeckung bei Schaltkreisen 5. Grundlagen des Schaltkreistestens: Testmustergenerierung und Fehlersimulation 6. Grundlagen des Softwaretestens 7. Formale Softwareverifikation auf Grundlage operationeller Semantik 8. Beweistheorien: Hoare Kalkül 9. Eigenschaftsprüfung und Äquivalenzprüfung für Schaltkreise 10. HW/SW-Coverifikation 								
Prüfungsformen	i.d.R. Bearbeitung von Übungsaufgaben und Fachgespräch oder mündliche Prüfung								

Literatur

- 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.
- Alexander Miczo: Digital Logic Testing and Simulation. Wiley 2004.
- Edmund M. Clarke, Orna Grumberg and Doron A. Peled: "Model Checking", The MIT Press, 1999
- Christel Baier and Joost-Pieter Katoen: "Principles of Model Checking", The MIT Press, 2008
- K. Apt, E.-R. Olderog: "Verification of Sequential and Concurrent Programs", Springer, 1991