

Modulhandbuch Online-Bachelor-Studiengang Medieninformatik

Redaktionsstand: 10. 06. 2013

Modul B12		Theoretische Informatik
Studiensemester	2.	
Credits	5	
Status	Pflichtmodul	
Häufigkeit des Angebotes	jedes Semester nach Bedarf der Hochschulen des VFH-Verbundes	
Lehrsprache	Deutsch	
Autor/in (verantwortliche Hochschule)	Prof. Dr. Friedhelm Seutter (Ostfalia HAW)	
Lerngebiet	Informatik, Theoretische Informatik	
Erworbene Kenntnisse, Fertigkeiten, Kompetenzen		
Formale, algorithmische, mathematische Kompetenzen	Wissen	Kennen und Wissen grundlegender Modelle und Methoden der Theoretischen Informatik und ihre Beziehungen untereinander.
	Verstehen	Verstehen der formalen Notationen, der ausgehend von Definitionen durch Sätze ausgedrückten Zusammenhänge und Beziehungen und der verwendeten Konstruktions- und Beweisideen. Verstehen der Automatenmodelle und der algebraischen und generierenden Konzepte zur Definition formaler Sprachen.
	Anwenden	Übertragen und Anwenden der auf formaler Ebene gewonnene Erkenntnisse auf Anwendungen der Praxis unter Berücksichtigung ihrer Beschränkungen.
	Analysieren	Analysieren konkreter Probleme, Reduktion und Abstraktion der Probleme auf das zur Lösung unbedingt Notwendige.
	Synthetisieren	Erstellen einer formalen Darstellung mittels Modellen und Methoden der Theoretischen Informatik zur Lösung des Problems.
	Evaluieren	Beschränkungen und Grenzen der Modelle und Methoden zur algorithmischen Berechnung von Lösungen verstehen und in Bezug auf ihre konkrete Anwendung bewerten und auswählen.
Analyse-, Design- und Realisierungskompetenzen	Wissen	
	Verstehen	
	Anwenden	
	Analysieren	
	Synthetisieren	
Technologische Kompetenzen	Wissen	
	Verstehen	
	Anwenden	
	Analysieren	
	Synthetisieren	
Fachübergreifende Kompetenzen	Wissen	
	Verstehen	
	Anwenden	
	Analysieren	
	Synthetisieren	
Methodenkompetenzen	Wissen	
	Verstehen	
	Anwenden	
	Analysieren	
	Synthetisieren	
Projektmanagement-Kompetenzen	Wissen	
	Verstehen	
	Anwenden	
	Analysieren	

Modulhandbuch Online-Bachelor-Studiengang Medieninformatik

Redaktionsstand: 10. 06. 2013

	Synthetisieren	
	Evaluieren	
Soziale Kompetenz und Selbstkompetenz	Wissen	
	Verstehen	
	Anwenden	
	Analysieren	
	Synthetisieren	
	Evaluieren	
Teilnahmevoraussetzungen		
zwingend	Module „Lineare Algebra“, „Einführung in die Informatik“, „Grundlagen der Programmierung 1“	
empfohlen	---	
Medien-/Lernform		
Multimedial aufbereitetes Online-Studienmodul zum Selbststudium mit zeitlich parallel laufender Online-Betreuung (E-Mail, Chat, Einsendeaufgaben u. a.) sowie Präsenzphasen.		
Arbeitsaufwand / Workload		Stunden
Selbststudium (online und offline, inkl. Prüfungsvorbereitung)		118
Pflichtpräsenzen an der Hochschule (inkl. Prüfungsteilnahme)		2
gemeinsame Online-Aktivitäten (Webkonferenzen, Foren u. ä.) und freiwillige Veranstaltungen an der Hochschule		30
Präsenzen		
Dauer	4 x 90 Minuten	
Präsenzinhalte	Zusammenfassung und Wiederholung ausgewählter Abschnitte aus dem Studienmodul, Klärung inhaltlicher Fragen, Besprechung von Übungsaufgaben, Klausurvorbereitung.	
Vermittlung der Präsenzinhalte	erfordert physische Anwesenheit	
Präsenzteilnahme ist	fakultativ	
Prüfung		
Prüfungsvorleistung	Zwei Einsendeaufgaben (min. 50% zum Bestehen)	
Teilleistungsnachweise	---	
Prüfungsform	Klausur (120 Minuten)	
Literatur	Sipser, M.: Introduction to the Theory of Computation. 2nd Edition. Thomson Course Technology, Boston 2006. ISBN 0-619-21764-2 Hopcroft, John E.; Motwani, Rajeev; Ullman, Jeffrey D.: Introduction to Automata Theory, Languages, and Computation. Second Edition. Addison-Wesley 2001.	
Sonstige Hinweise	---	
Studieninhalte Theoretische Informatik		
Zusammenfassung	<p>Das Studienmodul gibt eine Einführung in einige grundlegenden Modelle und Methoden der Theoretischen Informatik. Anhand von Automatenmodellen und von diesen analysierbaren formalen Sprachen werden die grundsätzlichen Fähigkeiten und Beschränkungen von Computern und Softwaresystemen untersucht. Dabei stehen insbesondere die Beziehungen zwischen den Automatenmodellen als analysierende Konzepte und den beschreibenden bzw. generierenden Konzepten für formale Sprachen im Vordergrund. Darüber hinaus wird die Frage diskutiert und beantwortet, ob gewisse Probleme überhaupt durch einen Computer oder ein Softwaresystem lösbar sind oder sich einer algorithmischen Berechnung verschließen. Die Studierenden sollen diese Modelle, Methoden und Konzepte kennen lernen und verstehen, sie in ihren fachlichen Kontext einordnen und in konkreten Problemen anwenden können.</p> <p>Die Modelle, Methoden und Konzepte und ihre Beziehungen untereinander werden teils informell erläutert, teils formal definiert bzw. hergeleitet. Für das Studium (insbesondere die Programmierausbildung) und die Praxis (insbesondere die Softwareentwicklung) können diese theoretischen Modelle</p>	

	<p>grundlegende Erkenntnisse und Hinweise zur Lösung diverser Probleme liefern.</p> <p>Computer und Softwaresysteme sind technische Systeme, die mit Hilfe mathematisch-formaler Modelle und Beschreibungen entwickelt und bedient werden. Auch neue Anwendungen sind auf dieser Basis zu konzipieren. Es ist deshalb unerlässlich, abstrakte Modelle und die darauf anzuwendenden Methoden mittels mathematisch-formaler Beschreibungen von Zuständen und Abläufen entwickeln, anpassen und anwenden zu können. Auch diese Kompetenzen sollen mit diesem Studienmodul eingeübt und vertieft werden.</p>
Überschriften der Kapitel/Lehreinheiten	<ol style="list-style-type: none">1. Formale Sprachen<ol style="list-style-type: none">1.1 Alphabete, Wörter und Sprachen1.2 Zusammenhang mit Programmiersprachen2. Endliche Automaten<ol style="list-style-type: none">2.1 Deterministische endliche Automaten2.2 Nichtdeterministische endliche Automaten3. Reguläre Sprachen<ol style="list-style-type: none">3.1 Reguläre Sprachen und Operationen3.2 Reguläre Ausdrücke3.3 Eigenschaften regulärer Sprachen4. Kontextfreie Sprachen<ol style="list-style-type: none">4.1 Kontextfreie Grammatiken4.2 Kellerautomaten4.3 Eigenschaften kontextfreier Sprachen5. Turingmaschinen und Berechenbarkeit<ol style="list-style-type: none">5.1 Deterministische Turingmaschinen5.2 Intuitiver Algorithmusbegriff5.3 Turing-Berechenbarkeit6. Entscheidbarkeit<ol style="list-style-type: none">6.1 Entscheidbare Probleme6.2 Das Halteproblem