

## Allgemein

<b>Studiengangskürzel</b>	20INB Version: 2
<b>Studiengang</b>	Informatik Bachelor Computer Science   Bachelor
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Abschluss</b>	Bachelor
<b>Erste Immatrikulation (gültig ab)</b>	2020
<b>Status</b>	Aktiv
<b>Regelstudienzeit in Semestern</b>	6 Semester
<b>Erforderliche Leistungspunkte</b>	180
<b>Studienmodus</b>	In Vollzeit studierbar
<b>Studienmodell</b>	Keine Angabe
<b>Für den Auslandsaufenthalt empfohlen</b>	5. Fachsemester
<b>Studiengangverantwortliche</b>	Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>
<b>Hinweise</b>	Diesen Studiengang finden Sie unter <a href="http://www.htwk-leipzig.de/inb">www.htwk-leipzig.de/inb</a> .

# Integrierter Studienablauf- und Prüfungsplan

Struktureinheit / Modul	ECTS	SWS (Vorlesung/Seminar/Übung/Praktikum) Prüfungs(vor)leistung (Gewicht, Dauer)					
		1. Sem.	2. Sem.	3. Sem.	4. Sem.	5. Sem.	6. Sem.
<b>Modellierung</b> Modelling C114.2 Pflichtmodul	8	4/0/2/0 PVB PVP <b>PK</b> 120 Min.					
<b>Digitaltechnik I</b> Digital Electronics I C752.1 Pflichtmodul	8	3/0/2/0 PVB <b>PK</b> 120 Min.					
<b>Grundlagen der Programmierung</b> Introduction to Programming C963.2 Pflichtmodul	7	2.5/0/0/2 <b>PM</b> 20 Min.					
<b>Mathematik für Informatiker I</b> Mathematics in Computer Science I N662.3 Pflichtmodul	7	4/0/2/0 PVB <b>PC</b> 120 Min.					
<b>Algorithmen und Datenstrukturen</b> Algorithms and Data Structures C300.2 Pflichtmodul	6		4/0/2/0 PVB PVP <b>PK</b> 120 Min.				
<b>Praktikum der Technischen Informatik</b> Computer Engineering Laboratory C405 Pflichtmodul	8		2/0/0/1 <b>PX<sup>1</sup></b> 33.33% 3 Mon. PVK	1/0/1/1 <b>PX<sup>1</sup></b> 33.33% 3 Mon. <b>PR<sup>1</sup></b> 33.33% 20 Min.			
<b>Anwendungsorientierte Programmierung</b> Applied Programming C680.2 Pflichtmodul	5		2/0/2/0 PVB <b>PJ</b> 4 Wo.				
<b>Digitaltechnik II</b> Digital Electronics II C947.1 Pflichtmodul	5		2/0/2/0 PVB <b>PK</b> 120 Min.				
<b>Mathematik für Informatiker II</b> Mathematics in Computer Science II N055.3 Pflichtmodul	5		2/0/2/0 PVB <b>PC</b> 90 Min.				
<b>Betriebssysteme und Rechnernetze</b> Operating Systems and Computer Networks C287.3 Pflichtmodul	6			4/0/0/2 <b>PK</b> 120 Min.			
<b>Softwaretechnik</b> Software Engineering C559.2 Pflichtmodul	5			2/0/1/1 PVJ PVT <b>PK</b> 120 Min.			
<b>Datenbanken</b> Database Systems C719.2 Pflichtmodul	5			2/0/1/1 PVJ <b>PK</b> 120 Min.			

Struktureinheit / Modul	ECTS	SWS (Vorlesung/Seminar/Übung/Praktikum) Prüfungs(vor)leistung (Gewicht, Dauer)					
		1. Sem.	2. Sem.	3. Sem.	4. Sem.	5. Sem.	6. Sem.
<b>Wissenschaftskommunikation in der Informatik</b> Writing and Presenting in Computer Science C902.1 Pflichtmodul	5			1/3/0/0 <b>PJ</b> <sup>1</sup> 60% 10 Wo. <b>PR</b> <sup>1</sup> 40% 15 Min.			
<b>Mathematik für Informatiker III</b> Mathematics in Computer Science III N417 Pflichtmodul	5			2/0/2/0 <b>PVB</b> <b>PK</b> 120 Min.			
<b>Softwareprojekt I</b> Software Engineering Project I C073 Pflichtmodul	5			1/0/0/1 <b>PJ</b> 13 Wo.			
<b>Fortgeschrittene Programmierung</b> Advanced Programming C393.2 Pflichtmodul	5			2/0/2/0 <b>PVB</b> <b>PK</b> 120 Min.			
<b>Rechnerarchitektur</b> Computer Architecture C827 Pflichtmodul	5			2/0/2/0 <b>PVR</b> <b>PM</b> 30 Min.			
<b>Automaten und formale Sprachen</b> Automata and Formal Languages C993.2 Pflichtmodul	5			2/0/2/0 <b>PVB</b> <b>PVP</b> <b>PK</b> 90 Min.			
<b>Softwareprojekt II</b> Software Engineering Project II C171 Pflichtmodul	5				0.5/0/0/1 <b>PJ</b> 13 Wo.		
<b>Grundlagen der Künstlichen Intelligenz</b> Foundations of Artificial Intelligence C622.2 Pflichtmodul	5				2/0/2/0 <b>PVB</b> <b>PK</b> 90 Min.		
<b>IT-Sicherheit</b> IT Security C799.2 Pflichtmodul	5				2/0/2/0 <b>PVP</b> <b>PK</b> 90 Min.		
<b>Praxisprojekt</b> Practical Project C222.3 Pflichtmodul	15					X <b>PVB</b> <b>PR</b> 30 Min. <b>TB</b> <sup>2</sup> 12 Wo.	
<b>Bachelormodul</b> Bachelor's Module C241.2 Pflichtmodul	15					X <b>PH</b> <sup>1</sup> 75% 3 Mon. <b>PV</b> <sup>1</sup> 25% 60 Min.	
<b>Hochschulkolleg - Überfachliche Kompetenzen</b>	5		5				

Struktureinheit / Modul	ECTS	SWS (Vorlesung/Seminar/Übung/Praktikum) Prüfungs(vor)leistung (Gewicht, Dauer)					
		1. Sem.	2. Sem.	3. Sem.	4. Sem.	5. Sem.	6. Sem.
<b>Englisch für Studium und Beruf (B2)</b> Academic and Vocational English (B2) F742.2 Pflichtmodul	3		0/2/0/0 <b>PR</b> <sup>1,3</sup> 25% 15 Min. <b>PK</b> <sup>1,3</sup> 75% 90 Min. PVC				
<b>Studium generale</b> General Studies U622 Pflichtmodul	2		2/0/0/0 <b>TB</b> <sup>2</sup>				
<b>Wahlpflicht</b>	<b>25</b>				<b>10</b>	<b>15</b>	
<b>Mikroprogrammierung und Mikroprozessoren</b> Microprogramming and Microprocessors C004.2 Wahlpflichtmodul	5				2/0/0/2 <b>PJ</b> 3 Mon.		
<b>Algorithmische Geometrie</b> Algorithmic Geometry C090 Wahlpflichtmodul	5				2/0/2/0 PVJ PVP <b>PK</b> 90 Min.		
<b>Computergrafik</b> Computer Graphics C121.2 Wahlpflichtmodul	5				2/0/2/0 PVC <b>PK</b> 120 Min.		
<b>Vertiefende Spezialvorlesung der Informatik 1</b> Special Topics in Computer Science C227.1 Wahlpflichtmodul	5				2/0/2/0 PVT <b>PK</b> 90 Min.		
<b>Semantic Web</b> Semantic Web C249.3 Wahlpflichtmodul	5				2/2/0/0 <b>PJ</b> 6 Wo.		
<b>Prozessautomatisierung</b> Automation of Technical Processes C383.2 Wahlpflichtmodul	5				2/0/0/2 PVB <b>PM</b> 30 Min.		
<b>Hardware-Entwurfstechnik</b> Hardware Design Tools C448.2 Wahlpflichtmodul	5				2/0/2/0 PVJ <b>PM</b> 30 Min.		
<b>e-Learning</b> e-Learning C585.2 Wahlpflichtmodul	5				2/0/2/0 PVJ <b>PM</b> 30 Min.		
<b>Datenbanken (Aufbaukurs)</b> Database Systems (Advanced Level) C720.2 Wahlpflichtmodul	5				2/0/2/0 PVT <b>PM</b> 30 Min.		
<b>Multimediale Webprogrammierung</b> Multimedia Web Programming C741.3 Wahlpflichtmodul	5				2/0/2/0 PVB <b>PK</b> 120 Min.		

Struktureinheit / Modul	ECTS	SWS (Vorlesung/Seminar/Übung/Praktikum) Prüfungs(vor)leistung (Gewicht, Dauer)					
		1. Sem.	2. Sem.	3. Sem.	4. Sem.	5. Sem.	6. Sem.
<b>Web Engineering</b> Web Engineering C864 Wahlpflichtmodul	5				2/2/0/0 PVT PK 90 Min.	2/2/0/0 PVT PK 90 Min.	
<b>Assemblerprogrammierung</b> Assembler Programming C960.2 Wahlpflichtmodul	5				2/2/0/0 PJ 6 Wo.		
<b>Einführung in ERP-Software (SAP)</b> Introduction to ERP Software (SAP) N450.2 Wahlpflichtmodul	5				2/0/2/0 PC 90 Min.		
<b>Diskrete Mathematik und Optimierung</b> Discrete Mathematics and Optimization N468 (N915) Wahlpflichtmodul	5				2/0/2/0 PK 90 Min.		
<b>Grundlagen der mobilen Robotik</b> Principles of Mobile Robotics C010 Wahlpflichtmodul	5					2/0/2/0 PVL <b>PH</b> <sup>1</sup> 50% 3 Mon. <b>PR</b> <sup>1</sup> 50% 20 Min.	
<b>Audio-Video-Kommunikation</b> Audio-Video Communication C064.2 Wahlpflichtmodul	5					0/2/0/2 PM 25 Min.	
<b>Constraint-Programmierung</b> Constraint Programming C160.3 Wahlpflichtmodul	5					2/0/2/0 PVB PK 120 Min.	
<b>Computeranimation</b> Computer Animation C182.2 Wahlpflichtmodul	5					2/0/2/0 PM 20 Min.	
<b>Grundlagen Internet-basierter Informationssysteme</b> Introduction to Internet Based Information Systems C247.2 Wahlpflichtmodul	5					2/0/2/0 PM 30 Min.	
<b>Thread-Programmierung</b> Thread Programming C259.3 (C529) Wahlpflichtmodul	5					2/0/2/0 PVT PK 120 Min.	
<b>Einführung in die virtuelle und erweiterte Realität (VR/AR)</b> Introduction to Virtual and Augmented Reality C493 (C384) Wahlpflichtmodul	5					2/0/2/0 <b>PJ</b> <sup>1</sup> 50% 6 Wo. <b>PR</b> <sup>1</sup> 50% 20 Min.	
<b>Dokumentbeschreibungssprachen</b> Document Description Languages C651.2 Wahlpflichtmodul	5					2/2/0/0 PVB PJ 6 Wo.	
<b>Mobile Computing</b> Mobile Computing C652.2 Wahlpflichtmodul	5					2/0/2/0 PC 120 Min.	
<b>Vertiefende Spezialvorlesung der Informatik 2</b> Special Topics in Computer Science C672.1 Wahlpflichtmodul	5					2/0/2/0 PVT PK 90 Min.	

Struktureinheit / Modul	ECTS	SWS (Vorlesung/Seminar/Übung/Praktikum) Prüfungs(vor)leistung (Gewicht, Dauer)					
		1. Sem.	2. Sem.	3. Sem.	4. Sem.	5. Sem.	6. Sem.
<b>Digitale Signal- und Bildverarbeitung</b> Digital Signal and Image Processing C763.1 Wahlpflichtmodul	5					2/0/2/0 PVB PK 120 Min.	
<b>Formale Methoden und Werkzeuge</b> Formal Methods and Tools C869.1 Wahlpflichtmodul	5					2/0/2/0 PVB PVP PK 120 Min.	
<b>Numerische Algorithmen</b> Numerical Mathematics N840 Wahlpflichtmodul	5					2/0/2/0 PVB PK 120 Min.	
<b>Einführung in die Betriebswirtschaftslehre</b> Introduction to Business Administration W233.2 (W861) Wahlpflichtmodul	5					2/2/0/0 PVR PK 90 Min.	
Summe SWS pro Semester:		21.5	25	25	22	21.5	0
Summe ECTS-Credits pro Semester:		30	30	30	30	30	30

<sup>1</sup> - Zu diesem Modul ist eine neuere Modulversion in Bearbeitung oder veröffentlicht.

<sup>1</sup> - Die Prüfungsleistung muss mit mindestens „ausreichend“ (4,0) bestanden sein.

<sup>2</sup> - Nicht benotete Prüfungsleistung, die bestanden sein muss.

<sup>3</sup> - Die Prüfungsleistung wird in einer Fremdsprache (siehe Lehrsprache) abgenommen.

PC - Prüfung am Computer | PH - Prüfung Hausarbeit | PJ - Prüfung Projektarbeit | PK - Prüfung Klausurarbeit | PM - Prüfung mündliches Fachgespräch | PR - Prüfung Referat | PV - Prüfung Verteidigung | PVB - Prüfungsvorleistung Beleg | PVC - Prüfungsvorleistung am Computer | PVJ - Prüfungsvorleistung Projektarbeit | PVK - Prüfungsvorleistung Klausurarbeit | PVL - Prüfungsvorleistung Laborarbeit | PVP - Prüfungsvorleistung Präsentation | PVR - Prüfungsvorleistung Referat | PVT - Prüfungsvorleistung Testat | PX - Prüfung Experiment | TB - Teilnahmebescheinigung | Min. - Minuten | Mon. - Monate | Std. - Stunden | Wo. - Wochen | SWS - Semesterwochenstunde

<b>Modul</b>	Softwareprojekt I Software Engineering Project I
<b>Modulnummer</b>	C073 Version: 0
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Andreas Both
<b>Dozierende</b>	Prof. Dr. rer. nat. Andreas Both
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	2 SWS (1 SWS Vorlesung   1 SWS Praktikum)
<b>Selbststudienzeit</b>	122 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Projektarbeit Prüfungsdauer: 13 Wochen   Wichtung: 100%
<b>Lehr- und Lernformen</b>	<ul style="list-style-type: none"> <li>- Begleitende Inputs werden als Vorlesung präsentiert</li> <li>- Die Projekte werden in Teams selbstorganisiert bearbeitet</li> <li>- ausgewählte Meetings und Zwischenstandspräsentationen werden durch die Betreuer organisiert und abgenommen</li> </ul>
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Vorstellung der Anforderungen</li> <li>- Teambildung</li> <li>- Erstellung einer Anforderungsspezifikation und einer Architekturvision mit Präsentationen an Meilensteinen</li> <li>- Erstellung einer produktiv einsetzbaren ersten Version der Software mit Präsentationen an Meilensteinen: erste Funktionalitäten sollten enthalten sein und prototypisch eine Vision für die Nutzungsoberfläche der gesamten Software vorhanden sein</li> </ul>
<b>Qualifikationsziele</b>	<p>Die Studierenden können sich an allen Phasen eines großen Softwareprojekts im Rahmen eines vorgegebenen agilen Vorgangsmodells (Scrum) beteiligen. Hierzu gehören insbesondere die folgenden Kompetenzen. Arbeitspakete können im Detail selbstständig geplant, termingerecht bearbeitet und dokumentiert werden. Sie können mit einem Dokumenten-Repository zum Versionsmanagement umgehen.</p> <p>Sie können für die konkreten Anforderungen einer zu erstellenden Anwendung Artefakte der Software-Entwicklung erstellen bzw. substantiell dazu beitragen.</p> <p>Die Studierenden erkennen Konflikte im Team und können Strategien zur Konfliktlösung anwenden. Selbstkompetenzen, wie Verbindlichkeit, Disziplin, Termintreue, Kompromissbereitschaft und die Übernahme von Verantwortung, werden projektdienlich entwickelt und eingesetzt.</p>
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Kompetenzen der Softwaretechnik und der Programmierung sollten soweit vorhanden sein, dass kleine Programme mit graphischer Benutzeroberfläche erstellt werden können

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- J. Ludewig, H. Lichter: "Software Engineering", dpunkt, in der aktuellen Auflage.</li> <li>- C. Rupp et al.: "UML 2 glasklar. Praxiswissen für die UML-Modellierung", Hanser, in der aktuellen Auflage.</li> <li>- H. Kellner: "Soziale Kompetenz für Ingenieure, Informatiker und Naturwissenschaftler", Hanser, 2006.</li> <li>- U. Vigenschow, B. Schneider: "Soft Skills für Softwareentwickler", dpunkt, in der aktuellen Auflage.</li> <li>- R. Pichler: "Scrum - Agiles Projektmanagement erfolgreich einsetzen", dpunkt, 2007.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Modellierung Modelling
<b>Modulnummer</b>	C114 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	8 ECTS-Punkte
<b>Workload</b>	240 Stunden
<b>Lehrveranstaltungen</b>	6 SWS (4 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	156 Stunden 28 Stunden Vorbereitung Lehrveranstaltung 28 Stunden E-Learning 84 Stunden Bearbeitung Prüfungsvorleistung 16 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg  Prüfungsvorleistung Präsentation
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Vorlesung - Übung - Bearbeiten von Problemen und Lösungsfindung,  Selbststudium anhand theoretischer und praktischer Übungsaufgaben
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	Modellierung und formale Darstellung von  - Daten durch Mengen, Mengenoperationen - Zusammenhängen durch Relationen, Funktionen, Äquivalenz- Ordnungsrelationen, Graphen - strukturierten Daten durch Wörter, Texte, Sprachen, Bäume, Signaturen, Terme, strukturelle Induktion, algebraische Strukturen - Eigenschaften und Anforderungen in Logiken (jeweils Syntax, Semantik, Folgern, Schließen) - Software-Schnittstellen durch abstrakte Datentypen - Abläufen und Berechnungen durch Zustandsübergangssysteme jeweils mit praktischen Modellierungsbeispielen
<b>Qualifikationsziele</b>	Die Studierenden können mathematische und logische Grundkonzepte zur Modellierung praktischer Aufgabenstellungen anwenden.  Sie können Anforderungen an Software und Systeme formal beschreiben und wissen, dass deren Korrektheit mit formalen Methoden nachweisbar ist.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Keine

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- U. Kastens, H. Kleine Büning: "Modellierung: Grundlagen und formale Methoden", Hanser, 2008.</li> <li>- M. Huth, M. Ryan: "Logic in Computer Science", Cambridge University Press, 2010.</li> <li>- U. Schöning: "Theoretische Informatik - kurzgefasst", Spektrum, in der aktuellen Auflage.</li> <li>- M. Broy, R. Steinbrüggen: "Modellbildung in der Informatik", Springer, 2004</li> </ul>
<b>Aktuelle Lehrressourcen</b>	Lehrmaterial und aktuelle Informationen: <a href="https://informatik.htwk-leipzig.de/schwarz">https://informatik.htwk-leipzig.de/schwarz</a>
<b>Hinweise</b>	regelmäßiges erfolgreiches Lösen der praktischen Übungsaufgaben (PVB) und 3 Kurzvorträge zu schriftlichen Übungsaufgaben (PVP)
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Pflichtmodul Medieninformatik   Bachelor (20MIB) Pflichtmodul Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Softwareprojekt II Software Engineering Project II
<b>Modulnummer</b>	C171 Version: 0
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Andreas Both
<b>Dozierende</b>	Prof. Dr. rer. nat. Andreas Both
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	1.50 SWS (0.50 SWS Vorlesung   1 SWS Praktikum)
<b>Selbststudienzeit</b>	129 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Projektarbeit Prüfungsdauer: 13 Wochen   Wichtung: 100%
<b>Lehr- und Lernformen</b>	<ul style="list-style-type: none"> <li>- begleitende Vorlesung mit Impulsreferaten</li> <li>- Abschlussveranstaltungen inklusive kleiner Produktmesse</li> <li>- Softwareentwicklung findet selbstorganisiert statt</li> <li>- bei Meetings und Zwischenstandspräsentationen hospitieren die Betreuer</li> </ul>
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Erstellung einer Anforderungsspezifikation und einer Architekturvision mit Präsentationen an mehreren Meilensteinen</li> <li>- Erstellung einer produktiv einsetzbaren Software mit Präsentationen an mehreren Meilensteinen</li> <li>- Poster-Abschlusspräsentation</li> <li>- Abschlusspräsentation als Vortrag</li> </ul>
<b>Qualifikationsziele</b>	<p>Studierende sind in der Lage ein in einem agilen Vorgehensmodell ein bestehendes Softwareentwicklungsprojekt fortzuführen und erfolgreich zu beenden, dass dem Kunden ein (zumindest partiell) funktionsfähiges Produkt ausgeliefert werden kann.</p> <p>Sie können fremden Quelltext lesen, darin Entwurfskonzepte erkennen sowie Änderungen durchführen. Sie erkennen selbstständig Schnittstellen zu den Arbeitspaketen anderer Teammitglieder, können die Probleme benennen und selbstständig Absprachen durchführen.</p> <p>Insbesondere sind sie in der Lage Teilmodule zu entwerfen und im Rahmen der Gesamtsoftware umzusetzen. Innerhalb des Projektkontexts beherrschen sie erfolgreich Strategien zur Qualitätssicherung, d.h. Fehlermanagement, Uni-Tests und Reviews. Die Qualität von Artefakten kann im Rahmen von Reviews beurteilt werden. Darüber hinaus werden im Projektkontext Probleme hinsichtlich der Planung und Durchführbarkeit erkannt sowie Maßnahmen vorgeschlagen.</p>
<b>Zulassungsvoraussetzung</b>	Keine

<b>Empfohlene Voraussetzungen</b>	<p>Der Besuch des Softwareprojekts I im vorherigen Semester ist dringend anzuraten, da die Teams und die Projekte fortgeführt werden. Andernfalls ist der Arbeitsaufwand für die Einarbeitung ungleich höher.</p> <p>Analog zu "Softwareprojekt I" werden auch hier hinreichend ausgeprägte Programmier- und Softwareentwicklungskompetenzen erwartet.</p>
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- J. Ludewig, H. Lichter: "Software Engineering", dpunkt, in der aktuellen Auflage.</li> <li>- C. Rupp et al.: "UML 2 glasklar. Praxiswissen für die UML-Modellierung", Hanser, in der aktuellen Auflage.</li> <li>- H. Kellner: "Soziale Kompetenz für Ingenieure, Informatiker und Naturwissenschaftler", Hanser, 2006.</li> <li>- U. Vigenschow, B. Schneider: "Soft Skills für Softwareentwickler", dpunkt, in der aktuellen Auflage.</li> <li>- R. Pichler: "Scrum - Agiles Projektmanagement erfolgreich einsetzen", dpunkt, 2007.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Praxisprojekt Practical Project
<b>Modulnummer</b>	C222 Version: 3
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr.-Ing. Thomas Kudraß <a href="mailto:thomas.kudrass@htwk-leipzig.de">thomas.kudrass@htwk-leipzig.de</a>
<b>Dozierende</b>	
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	15 ECTS-Punkte
<b>Workload</b>	450 Stunden
<b>Lehrveranstaltungen</b>	0 SWS
<b>Selbststudienzeit</b>	0 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Referat Prüfungsdauer: 30 Minuten   Wichtung: 100%  Teilnahmebescheinigung Prüfungsdauer: 12 Wochen   Wichtung: 0%   nicht benotet   nicht kompensierbar
<b>Lehr- und Lernformen</b>	Am Beginn des Projekts arbeitet die Praxisstelle einen Praktikumsplan aus, der vom Studierenden mit dem betreuenden Hochschullehrer diskutiert wird. Während der Betreuung durch die Hochschule erfolgt eine regelmäßige individuelle Besprechung des Arbeitsfortschritts im Rahmen von Konsultationen, in Verbindung mit einer Orientierung auf den zu erstellenden Bericht sowie die Präsentation.
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- themenspezifisch
<b>Qualifikationsziele</b>	Ziele: Das Praxisprojekt wird in einem Unternehmen oder in einer anderen Einrichtung der Berufspraxis abgeleistet. Es dient der Vermittlung praktischer Erfahrungen und Fähigkeiten zur Ergänzung der theoretischen Kenntnisse.  Kompetenzen: Der Studierende soll den Einsatz seiner Fachkenntnisse in der Praxis üben, praktische Aufgaben und Zusammenhänge abstrahieren lernen und seine Kommunikations- und Teamfähigkeit ausbauen. Abschließend soll er seine Fähigkeit unter Beweis stellen, die eigene Tätigkeit im Praxisprojekt kompakt im Rahmen eines Vortrages darzustellen.  Einbindung in die Berufsvorbereitung: Das Praxisprojekt dient der unmittelbaren Berufsvorbereitung. Es kann sehr gut zu einer persönlichen Sondierung und Kontaktherstellung zu potenziellen späteren Arbeitgebern genutzt werden.
<b>Zulassungsvoraussetzung</b>	Festlegung durch Prüfungsordnung und Praktikumsordnung
<b>Empfohlene Voraussetzungen</b>	Keine
<b>Literaturhinweise</b>	- themenspezifisch
<b>Aktuelle Lehrressourcen</b>	keine

<b>Hinweise</b>	<p>Workload: 450h entsprechen 12 Wochen Tätigkeit auf einer Praxisstelle</p> <p>Prüfungsvorleistung Beleg (PVB): Praktikumsbericht des Studenten Tätigkeitsnachweis der Praxisstelle</p>
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Bachelormodul Bachelor's Module
<b>Modulnummer</b>	C241 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Studiendekan (INB, INM)
<b>Dozierende</b>	
<b>Sprache(n)</b>	Deutsch in „Bachelorarbeit“  Englisch in „Bachelorarbeit“  Deutsch in „Bachelorkolloquium“
<b>ECTS-Leistungspunkte</b>	15 ECTS-Punkte
<b>Workload</b>	450 Stunden 360 Stunden in „Bachelorarbeit“ 90 Stunden in „Bachelorkolloquium“
<b>Lehrveranstaltungen</b>	0 SWS 0 SWS in „Bachelorarbeit“ 0 SWS in „Bachelorkolloquium“
<b>Selbststudienzeit</b>	450 Stunden 360 Stunden in „Bachelorarbeit“ 90 Stunden in „Bachelorkolloquium“
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Hausarbeit Prüfungsdauer: 3 Monate   Wichtung: 75%   nicht kompensierbar in „Bachelorarbeit“  Prüfung Verteidigung Prüfungsdauer: 60 Minuten   Wichtung: 25%   nicht kompensierbar in „Bachelorkolloquium“
<b>Lehr- und Lernformen</b>	„Bachelorarbeit“: Die Bachelorarbeit wird selbstständig bearbeitet und die resultierende Hausarbeit selbstständig verfasst. Der Prozess wird durch Konsultationen begleitet.  „Bachelorkolloquium“: -
<b>Medienform</b>	„Bachelorarbeit“: keine Angabe
<b>Lehrinhalte/Gliederung</b>	„Bachelorarbeit“: - themenspezifisch  „Bachelorkolloquium“: - themenspezifisch

<b>Qualifikationsziele</b>	<p><b>Bachelorarbeit:</b> Mit der Bachelorarbeit zeigt der Student, dass er in der Lage ist, ein umfangreiches Problem seines Fachgebiets innerhalb einer vorgegebenen Frist mit üblichen fachspezifischen Methoden zu bearbeiten und dazu eine schriftliche wissenschaftliche Arbeit zu verfassen. Das Thema wird durch einen Professor (den Betreuer der Arbeit) festgelegt.</p> <p><b>Bachelorkolloquium:</b> Im Bachelorkolloquium stellt der Student die Fähigkeit unter Beweis, Inhalt, Methodik und Ergebnisse seiner Arbeit objektiv und ansprechend zu präsentieren und in der wissenschaftlichen Diskussion zu verteidigen</p>
<b>Zulassungsvoraussetzung</b>	Festlegung durch Prüfungsordnung
<b>Empfohlene Voraussetzungen</b>	Keine
<b>Literaturhinweise</b>	<p>„<b>Bachelorarbeit</b>“: - themenspezifisch</p> <p>„<b>Bachelorkolloquium</b>“: - themenspezifisch</p>
<b>Aktuelle Lehrressourcen</b>	<p>„<b>Bachelorarbeit</b>“: keine</p> <p>„<b>Bachelorkolloquium</b>“: keine</p>
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Betriebssysteme und Rechnernetze Operating Systems and Computer Networks
<b>Modulnummer</b>	C287 Version: 3
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr.-Ing. Jean-Alexander Müller <a href="mailto:jean-alexander.mueller@htwk-leipzig.de">jean-alexander.mueller@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr.-Ing. Jean-Alexander Müller <a href="mailto:jean-alexander.mueller@htwk-leipzig.de">jean-alexander.mueller@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch in "Betriebssysteme"  Deutsch in "Rechnernetze"
<b>ECTS-Leistungspunkte</b>	6 ECTS-Punkte
<b>Workload</b>	180 Stunden 90 Stunden in "Betriebssysteme" 90 Stunden in "Rechnernetze"
<b>Lehrveranstaltungen</b>	6 SWS (4 SWS Vorlesung   2 SWS Praktikum) 3 SWS (2 SWS Vorlesung   1 SWS Praktikum) in "Betriebssysteme" 3 SWS (2 SWS Vorlesung   1 SWS Praktikum) in "Rechnernetze"
<b>Selbststudienzeit</b>	94 Stunden 46 Stunden in "Betriebssysteme" 48 Stunden in "Rechnernetze"
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	<b>Betriebssysteme:</b> - Seminaristische Vorlesung - Übungen zu Theorie und praktischen Fertigkeiten im Computerpool - Übungsaufgaben für das Selbststudium  <b>Rechnernetze:</b> - Seminaristische Vorlesung - Übungen zu Theorie und praktischen Fertigkeiten im Computerpool - Übungsaufgaben für das Selbststudium
<b>Medienform</b>	<b>Betriebssysteme:</b> - Vorlesungen kombinieren vorbereitete Präsentationen und Erarbeitung von Themen an der Tafel - Übungsaufgaben maßgeblich aus Standardwerken des Lehrgebiets  <b>Rechnernetze:</b> - Vorlesungen kombinieren vorbereitete Präsentationen und Erarbeitung von Themen an der Tafel - Übungsaufgaben maßgeblich aus Standardwerken des Lehrgebiets

<b>Lehrinhalte/Gliederung</b>	<p><b>Betriebssysteme:</b></p> <ul style="list-style-type: none"> <li>- Aufgabenstellung und Begriffsbestimmung</li> <li>- Entwicklung von Betriebssystemen</li> <li>- Klassifikation und Methodik <ul style="list-style-type: none"> <li>- Prozesse: Konzept, Beschreibung, Kontrolle von Prozessen</li> <li>- Speicherverwaltung</li> <li>- Interprozesskommunikation: Signale, Pipes, Sockets, System V IPC (Message Queues, Semaphore, Shared Memory)</li> <li>- Prozesskoordination: Concurrency, kritische Bereiche, Lösungsansätze</li> <li>- Scheduling: Typen, Bursts, Prozess-Scheduling, Schedulingalgorithmen</li> <li>- Virtualierungskonzepte</li> <li>- Dateisysteme</li> <li>- Sicherheitmechanismen</li> </ul> </li> <li>- PC-Betriebssysteme als Beispiel <ul style="list-style-type: none"> <li>- Prozesse, Dateisysteme, Nutzer</li> <li>- Kommandoprozeduren unter Linux</li> <li>- parallele Prozesse unter Linux</li> <li>- einfache Formen der Kommunikation paralleler Prozesse</li> <li>- praktische Übungen zur Programmierung von Kommandoprozeduren und parallelen Prozessen</li> </ul> </li> </ul> <p><b>Rechnernetze:</b></p> <ul style="list-style-type: none"> <li>- Einführung in Netzwerktechnologien und Strukturen <ul style="list-style-type: none"> <li>- Datacenter / Vernetzung in Rechenzentren</li> <li>- Lokale Netze bis zum Intranet</li> <li>- Das Internet und andere Weltverkehrsnetze</li> <li>- Überblick zu Mobil- und Zugangsnets</li> </ul> </li> <li>- Architektur und Grundprinzipien <ul style="list-style-type: none"> <li>- Paketvermittlung, Referenzmodelle und Betriebsverfahren</li> <li>- Scheduling und Planung</li> <li>- Direktverbindungsnetze</li> <li>- Vermittlungsprinzipien, Routingverfahren</li> <li>- Tunnel, Overlay</li> <li>- Sicherheitsaspekte</li> </ul> </li> <li>- Technologien <ul style="list-style-type: none"> <li>- Internet Protocol (v4, v6, vX)</li> <li>- IEEE 802-Technologien</li> <li>- Virtualisierung, SDN, OpenFlow</li> <li>- Carrier Ethernet, GMPLS</li> </ul> </li> </ul>
<b>Qualifikationsziele</b>	<p><b>Betriebssysteme:</b> Die Studierenden können Grundkonzepte von modernen Betriebssystemen formal und sprachlich korrekt beschreiben und sind in der Lage, sie auf PC-Plattformen anzuwenden und nutzbar zu machen. Sie können selbstständig und mit angemessenen Mitteln Betriebssysteme auf PC-Plattformen installieren und anpassen. Sowohl die Erstellung von Unix-spezifischen Anwendungsprogrammen unter Einsatz der Unix-API wie auch die Programmierung von Kommandoprozeduren kann selbstständig unter Nutzung der vorhandenen Systemdokumentationen durchgeführt werden.</p> <p><b>Rechnernetze:</b> Es besteht detailliertes, anwendungsfähiges Fachwissen auf dem Gebiet der Netzwerktechnologien, Strukturen und deren Grundprinzipien. Aufsetzend auf dem Verständnis der Grundprinzipien sowie der erworbenen praktischen Fähigkeiten sind sie in der Lage veränderte Methoden und Trends zu erkennen und deren Potential gegenüber etablierten Technologien zu ermitteln.</p>
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Fertigkeiten in der Programmierung (derzeit C-Programmierung)

<p><b>Literaturhinweise</b></p>	<p><b>Betriebssysteme:</b></p> <ul style="list-style-type: none"> <li>- W. Stallings: Operating Systems. Prentice Hall, New Jersey, 2003</li> <li>- Viel Erfolg bei Ihrem Förderantrag und gute Rekonvaleszenz.</li> <li>- Silberschatz: Operating System Concepts, 9nd. Wiley, 2012</li> <li>- M. Hailperin: "<a href="#">Operating Systems an Middleware, Supporting Controlled Interception</a>", CC BY-SA 3.0, Rev 1.3</li> <li>- J. Plötner, S. Wendzel: "<a href="#">Linux - Das umfassende Handbuch</a>", Rheinwerk Computing, 2012</li> </ul> <p><b>Rechnernetze:</b></p> <ul style="list-style-type: none"> <li>- P. L. Dordal: "<a href="#">An Introduction to Computer Networks</a>", CC BY-NC-ND 3.0, 2019.</li> <li>- A. S. Tanenbaum, D. J. Wetherall: „Computer Networks“, Prentice Hall, 5. Auflage, 2010.</li> <li>- K. R. Fall, W. R. Stevens: "TCP/IP Illustrated volume 1: The Protocols", Addison-Wesley, 2011.</li> <li>- L. L. Peterson, B. S. Davie: "Computer Networks: A Systems Approach", Morgan Kaufmann, 5. Auflage, 2011.</li> <li>- T. Nadeu, K. Gray: "SDN: Software Defined Networks", O'Reilly, 2013.</li> <li>- „Ethernet“, Heise Verlag, 2008.</li> </ul>
<p><b>Aktuelle Lehrressourcen</b></p>	<p><b>Betriebssysteme:</b> keine</p> <p><b>Rechnernetze:</b> keine</p>
<p><b>Hinweise</b></p>	<p>Keine Angabe</p>
<p><b>Verwendbarkeit</b></p>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<p><b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b></p>	

<b>Modul</b>	Algorithmen und Datenstrukturen Algorithms and Data Structures
<b>Modulnummer</b>	C300 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Karsten Weicker <a href="mailto:karsten.weicker@htwk-leipzig.de">karsten.weicker@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Karsten Weicker <a href="mailto:karsten.weicker@htwk-leipzig.de">karsten.weicker@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	6 ECTS-Punkte
<b>Workload</b>	180 Stunden
<b>Lehrveranstaltungen</b>	6 SWS (4 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	96 Stunden 60 Stunden Vorbereitung Lehrveranstaltung 12 Stunden Bearbeitung Prüfungsvorleistung 24 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg Prüfungsvorleistung Präsentation
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	<p>Die Vorlesung wird gemäß des Flipped-Classroom-Konzepts durchgeführt. Dies bedeutet, dass 2 SWS Vorlesungszeit sowie weitere 22h der Selbststudienzeit als Lehrvideos bereit gestellt werden. Die Präsenztermine dienen der Vertiefung, Besprechung von Lösungen und Diskussion von Problemen.</p> <p>Über ein Punktesystem ergibt sich die Prüfungsvorleistung aus</p> <ul style="list-style-type: none"> <li>- begleitenden kleinen Übungsaufgaben zu den Lehrvideos,</li> <li>- gegenseitiger Peer-Begutachtung dieser Übungsaufgaben,</li> <li>- 14-tägigen in kooperativen Lerngruppen zu bearbeitende Aufgaben sowie deren Präsentation in den Übungen und</li> <li>- einem von drei möglichen Programmierbelegen.</li> </ul> <p>Bei den Übungen gemäß der kooperativen Lerngruppen bearbeiten 3-4 Studierende gemeinsam die Aufgaben und melden welche Aufgaben vortragsbereit sind. Ist eine Aufgabe gemeldet, muss jedes Gruppenmitglied in der Lage sein, die ausgegebene oder eine ähnliche Aufgabe an der Tafel zu bearbeiten. Aus der Präsentation an der Tafel ergeben sich die Punkte für alle Mitglieder der Gruppe.</p> <p>Die Klausur findet als Two-Stage-Exam statt, bei welchem im letzten Drittel der Prüfung Kommunikation in Kleingruppen erlaubt ist. Ziel der kooperativen Übungen und des Two-Stage-Exams ist die Beförderung der Problemlösekompetenz durch Teamwork und Kommunikation.</p>
<b>Medienform</b>	keine Angabe

<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Grundlagen</li> <li>- Einfache Suchalgorithmen (Listen und Felder)</li> <li>- Bäume (Suchbäume, AVL-Bäume, optimale Suchbäume)</li> <li>- Sortieren (Quicksort, Heapsort, Mergesort)</li> <li>- Hashing (extern, offen, Brent's Algorithmus)</li> <li>- Graphenalgorithmen (minimaler Spannbaum, kürzeste Wege, Rundreiseproblem)</li> <li>- Entwurfsparadigmen: Divide-and-Conquer, dynamisches Programmieren, Backtracking, Greedy</li> </ul>
<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss des Moduls, haben die Studierenden die behandelten Standarddatenstrukturen und -algorithmen so weit verstanden, dass sie diese am Beispiel nachvollziehen können. Ferner können sie einfache Algorithmen bzgl. der Laufzeit und des Speicherbedarfs analysieren - u.a. unter Verwendung eines Mastertheorems. Algorithmen können in einem Anwendungsszenario implementiert werden. Laufzeitmessungen können theoretischen Resultaten gegenübergestellt werden. Für einfache Aufgabenstellungen können die Studierenden eigene Algorithmen entwickeln. Die Studierenden sind im Rahmen der fachlichen Lehrinhalte in der Lage, gemeinsam algorithmische Lösungen für Probleme zu diskutieren und zu erarbeiten.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Keine
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- K. Weicker, N. Weicker: "Algorithmen und Datenstrukturen", SpringerVieweg, 2013.</li> <li>- T. Ottmann, P. Widmayer: "Algorithmen und Datenstrukturen", Spektrum, in der aktuellen Auflage.</li> <li>- T. H. Cormen et al.: "Algorithmen - Eine Einführung", Oldenbourg, in der aktuellen Auflage.</li> <li>- R. Sedgewick: "Algorithmen in Java", Addison-Wesley, in der aktuellen Auflage.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Die Selbststudienzeit setzt sich zusammen aus 22 h für die Lehrvideos, 60 h für die Bearbeitung der Übungsaufgaben und 14 h Vorbereitung für die Prüfung.
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Fortgeschrittene Programmierung Advanced Programming
<b>Modulnummer</b>	C393 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Vorlesung - Übung - E-Learning (automatische Bewertung eines Teiles der Hausaufgaben)
<b>Medienform</b>	- Tafelanschrieb - Skript
<b>Lehrinhalte/Gliederung</b>	- algebraische Datentypen, Pattern Matching, Termersetzung - Funktionen (polymorph getypt, von höherer Ordnung), Lambda-Kalkül, Rekursionsmuster (map, fold) - Typklassen, Interfaces, Unit-Tests, automatische Testfallerzeugung - funktionales Programmieren in modernen multiparadigmatischen Sprachen - Bedarfsauswertung, unendliche Datenstrukturen, Iteratoren - Codequalität, Code smells, Refaktorisierung
<b>Qualifikationsziele</b>	Die Studierenden haben fortgeschrittene Konzepte der Programmierung sowie ihre Ausprägungen in verschiedenen Programmiersprachen erlernt. Sie können diese Konzepte bei konkreten Programmieraufgaben anwenden.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Keine
<b>Literaturhinweise</b>	- J. Waldmann: „How I Teach Functional Programming“, Proc. WFLP 2017. - M. Naftalin, P. Wadler: "Java generics and Collections", O'Reilly, 2006. - B. O'Sullivan, D. Stewart, J. Goerzen: "Real World Haskell", O'Reilly, 2008.
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistung: Regelmäßiges und erfolgreiches Bearbeiten von Übungsaufgaben

<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Pflichtmodul Medieninformatik   Bachelor (20MIB) Pflichtmodul Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Praktikum der Technischen Informatik Computer Engineering Laboratory
<b>Modulnummer</b>	C405 Version: 0
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	2 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a> Dozentin/Dozent in: "Hardwarepraktikum I", "Hardwarepraktikum II", "Systemnahe Programmierung"  Prof. Dr. rer. nat. Hanna Brodowsky <a href="mailto:hanna.brodowsky@htwk-leipzig.de">hanna.brodowsky@htwk-leipzig.de</a> Dozentin/Dozent in: "Elektrotechnische Grundlagen"
<b>Sprache(n)</b>	Deutsch in "Hardwarepraktikum I"  Deutsch in "Hardwarepraktikum II"  Deutsch in "Systemnahe Programmierung"  Deutsch in "Elektrotechnische Grundlagen"
<b>ECTS-Leistungspunkte</b>	8 ECTS-Punkte
<b>Workload</b>	240 Stunden 60 Stunden in "Hardwarepraktikum I" 60 Stunden in "Hardwarepraktikum II" 60 Stunden in "Systemnahe Programmierung" 60 Stunden in "Elektrotechnische Grundlagen"
<b>Lehrveranstaltungen</b>	6 SWS (3 SWS Vorlesung   1 SWS Übung   2 SWS Praktikum) 1 SWS (1 SWS Praktikum) in "Hardwarepraktikum I" 1 SWS (1 SWS Praktikum) in "Hardwarepraktikum II" 2 SWS (1 SWS Vorlesung   1 SWS Übung) in "Systemnahe Programmierung" 2 SWS (2 SWS Vorlesung) in "Elektrotechnische Grundlagen"
<b>Selbststudienzeit</b>	156 Stunden 46 Stunden in "Hardwarepraktikum I" 46 Stunden in "Hardwarepraktikum II" 32 Stunden in "Systemnahe Programmierung" 32 Stunden in "Elektrotechnische Grundlagen"
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Klausurarbeit in "Elektrotechnische Grundlagen"
<b>Prüfungsleistung(en)</b>	Prüfung Experiment Prüfungsdauer: 3 Monate   Wichtung: 33.33%   nicht kompensierbar in "Hardwarepraktikum I"  Prüfung Experiment Prüfungsdauer: 3 Monate   Wichtung: 33.33%   nicht kompensierbar in "Hardwarepraktikum II"  Prüfung Referat Prüfungsdauer: 20 Minuten   Wichtung: 33.33%   nicht kompensierbar in "Systemnahe Programmierung"

<b>Lehr- und Lernformen</b>	<p><b>Hardwarepraktikum I:</b> Nach einer einführenden Vorlesung ist ein komplexes Experiment selbstständig vorzubereiten und zu einem gegebenen Abgabetermin unter Beantwortung von Fragen vorzuführen. Die Vorbereitung findet im Labor statt, es sind regelmäßig Zwischenschritte abzurechnen.</p> <p><b>Hardwarepraktikum II:</b> Nach einer einführenden Vorlesung ist ein komplexes Experiment selbstständig vorzubereiten und zu einem gegebenen Abgabetermin unter Beantwortung von Fragen vorzuführen. Die Vorbereitung findet im Labor statt, es sind regelmäßig Zwischenschritte abzurechnen.</p> <p><b>Systemnahe Programmierung:</b> Unter Anleitung werden die Grundlagen der systemnahen Programmierung durch praktische Beispielapplikationen im Labor erarbeitet. Mit Hilfe des gewonnenen Wissen wird eine komplexe Projektaufgabe über einen längeren Zeitraum abgearbeitet. In regelmäßigen Konsultationen werden Teilaufgaben abgerechnet, bei einer abschließenden Präsentation wird das Projekt bewertet.</p> <p><b>Elektrotechnische Grundlagen:</b> Experimentalvorlesung mit integrierten Übungen</p> <p>16 Stunden werden als Blockveranstaltung in der ersten Woche gehalten</p>
<b>Medienform</b>	<p><b>Hardwarepraktikum I:</b> keine Angabe</p> <p><b>Hardwarepraktikum II:</b> keine Angabe</p> <p><b>Systemnahe Programmierung:</b> keine Angabe</p> <p><b>Elektrotechnische Grundlagen:</b> keine Angabe</p>
<b>Lehrinhalte/Gliederung</b>	<p><b>Hardwarepraktikum I:</b></p> <ul style="list-style-type: none"> <li>- Analoge und digitale Messtechnik</li> <li>- Kennlinien von verschiedenen Diodenarten</li> <li>- Transistor als Schalter</li> <li>- Signalausbreitung auf Kabeln</li> </ul> <p><b>Hardwarepraktikum II:</b></p> <ul style="list-style-type: none"> <li>- Kombinatorische Logik und Flipflops</li> <li>- Mikrocontroller in Steuerungsanwendungen</li> <li>- Leistungsaufnahme von Prozessoren</li> <li>- Schnittstellen und Kommunikation</li> <li>- Automatenentwurf in einer HDL</li> </ul> <p><b>Systemnahe Programmierung:</b></p> <ul style="list-style-type: none"> <li>- Einführung mit historischer Rechentechnik</li> <li>- Mikroprozessoren und Mikroprozessorsysteme</li> <li>- Programmiermodell und Instruktionen</li> <li>- Programmieren ganzzahliger Arithmetik</li> <li>- Werkzeuge der Maschinenprogrammierung</li> </ul> <p><b>Elektrotechnische Grundlagen:</b></p> <ul style="list-style-type: none"> <li>- Stromkreise und Bauelemente</li> <li>- Funktionsweise von Halbleiterbauelementen</li> <li>- Analogschaltungen mit Halbleiterbauelementen</li> <li>- Logikschaltungen</li> </ul>

<b>Qualifikationsziele</b>	<p>Die Studenten haben ein grundsätzliches Verständnis für die Funktionen passiver und aktiver Bauelemente sowie digitaler Schaltkreise und können mit geeigneten Messmitteln deren Eigenschaften darstellen und bewerten. Die problembezogene Auswahl und Anwendung von Verfahren der computergestützten Messtechnik und von Messmitteln wie Multimeter und Oszilloskop wird von ihnen bei typischen Standardaufgaben beherrscht. Die Studierenden können komplexe Aufgabenstellungen analysieren und Lösungsabläufe planen und ausführen. Neben allgemeinen Kompetenzen wie der zeitlichen Ablaufplanung des Praktikums und der sprachlichen Präsentation der Resultate werden manuelle Fertigkeiten beim Schaltungsaufbau sowie die Verknüpfung von technischem und theoretischem Wissen gefördert.</p> <p>Die Studierenden sind in der Lage, Programmiermodell und Ausführungslogik von Mikroprozessoren zu beschreiben und die Ausdrucksmittel dieser Architekturen zur Lösung systemnaher Aufgabenstellungen adäquat einzusetzen. Algorithmen der Ganzzahlarithmetik und zur Manipulation von Datenstrukturen können auf die Systemarchitektur abgebildet und mittels einer einfachen Entwicklungsumgebung implementiert werden.</p>
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Geübter Umgang mit den physikalischen Grundgrößen und ihren Maßeinheiten sowie ihre Anwendung auf Gleichstromkreise. Verständnis zu allgemeinen Hardwaregrundlagen insbesondere zu digitalen Schaltungen. Fähigkeit zum Entwurf von Schaltnetzen, praktische Erfahrungen mit einer anwendungsorientierten Programmiersprache.
<b>Literaturhinweise</b>	<p><b>Hardwarepraktikum I:</b>        - Aufgabenspezifische Versuchsanleitungen zu den Praktikumsversuchen        - Schiffmann, Schmitz: Technische Informatik 1: Grundlagen der Digitalen Elektronik, Springer 2004 oder aktueller</p> <p><b>Hardwarepraktikum II:</b>        - Aufgabenspezifische Versuchsanleitungen zu den Praktikumsversuchen        - Schiffmann, Schmitz: Technische Informatik 1: Grundlagen der Digitalen Elektronik, Springer 2004 oder aktueller</p> <p><b>Systemnahe Programmierung:</b>        - Schiffmann, Schmitz: Technische Informatik 2: Grundlagen der Computertechnik, Springer 2005 oder aktueller</p> <p><b>Elektrotechnische Grundlagen:</b>        - Schiffmann, Schmitz, Weiland: Technische Informatik, Übungsbuch, Springer, in der aktuellen Auflage.        - H. Lindner: „Physik für Ingenieure“, Vieweg, in der aktuellen Auflage.        - G. Koß, W. Reinhold: „Lehr- und Übungsbuch Elektronik“, Fachbuchverlag Leipzig, in der aktuellen Auflage.        - R. Paul: „Elektrotechnik für Informatiker“, Teubner, in der aktuellen Auflage.</p>
<b>Aktuelle Lehrressourcen</b>	<p><b>Hardwarepraktikum I:</b>        keine</p> <p><b>Hardwarepraktikum II:</b>        keine</p> <p><b>Systemnahe Programmierung:</b>        keine</p> <p><b>Elektrotechnische Grundlagen:</b>        keine</p>
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Pflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Softwaretechnik Software Engineering
<b>Modulnummer</b>	C559 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Karsten Weicker <a href="mailto:karsten.weicker@htwk-leipzig.de">karsten.weicker@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Karsten Weicker <a href="mailto:karsten.weicker@htwk-leipzig.de">karsten.weicker@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   1 SWS Übung   1 SWS Praktikum)
<b>Selbststudienzeit</b>	94 Stunden 30 Stunden Vorbereitung Lehrveranstaltung 44 Stunden Bearbeitung Prüfungsvorleistung 20 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Projektarbeit Prüfungsvorleistung Testat
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Die Vorlesung findet nach dem Konzept des Inverted Classrooms statt.  Neben der Vorlesung bearbeiten die Studierenden typische Anwendungs- und Klausuraufgaben. In einem kleinen Lehrveranstaltungsbegeleitenden Projekt werden die typischen Aktivitäten/Phasen der Softwareentwicklung angewandt.  Lehrvideos sind über den Mediaserver der HTWK nach Anmeldung abrufbar.
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Überblick über den Software-Lebenszyklus, Gesetzmäßigkeiten des Software Engineering - Anforderungsspezifikation (UML, GUI-Prototypen) - Entwurf (Architekturprinzipien, Überblick über Software-Architekturen, Grob- und Feinentwurf, Entwurfsmuster) - Implementierung (Programmierrichtlinien, Unit-Tests, Refactoring, Versionsmanagement) - Projektmanagement (agile Software-Entwicklung, Prozessmodelle, Kostenschätzung, Aspekte der Planung, Reengineering-Projekte)
<b>Qualifikationsziele</b>	Die Studierenden können Dokumente aus den unterschiedlichen Phasen der Softwareentwicklung lesen, für kleine Projekte selbst erstellen und kritisch hinsichtlich der Qualität bewerten. Sie beherrschen Notationen und Werkzeuge der UML-Modellierung und der Anforderungsspezifikation.  Ferner können sie existierende Projekte hinsichtlich der Software-Architektur untersucht sowie für kleine Projekte selbige entwickeln und umsetzen. Werkzeuge zum Testen von Software, Refactoring, Versionsmanagement und Quelltextdokumentation werden beherrscht
<b>Zulassungsvoraussetzung</b>	Keine

<b>Empfohlene Voraussetzungen</b>	Programmierkompetenzen sollten soweit vorhanden sein, dass kleine Programme mit graphischer Benutzeroberfläche erstellt werden können.
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- J. Ludewig, H. Lichter: "Software Engineering", dpunkt, in der aktuellen Auflage.</li> <li>- A. Endres, D. Rombach: "A Handbook of Software and Systems Engineering", Pearson, 2003.</li> <li>- C. Rupp et al.: "UML 2 glasklar. Praxiswissen für die UML-Modellierung", Hanser, in der aktuellen Auflage.</li> <li>- G. Starke: "Effektive Software-Architekturen", Hanser, in der aktuellen Auflage.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistungen: wöchentliche Bearbeitung von Aufgaben im Seminar und erfolgreiche Bearbeitung eines Anwendungsprojekts in kleinen Teams
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Grundlagen der Künstlichen Intelligenz Foundations of Artificial Intelligence
<b>Modulnummer</b>	C622 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 28 Stunden Selbststudium 56 Stunden Bearbeitung Prüfungsvorleistung 10 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Vorlesung - Übung - Bearbeiten von Problemen und Lösungsfindung, - Selbststudium anhand theoretischer und praktischer Übungsaufgaben
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Intelligente Agenten: Aktionen und Verhalten, Struktur und Umgebungen - Repräsentation von Wissen: Logik, Regeln - Deduktion und Problemlösen - Ausblick logische Programmierung, Resolution - Suchverfahren - Wissensbasiertes Planen - Ausgewählte Beispiele: Robotik, Spiele und Diagnosesysteme - Ausblick nichtklassische Logiken: nichtmonotonen Schließen, Temporallogik, Fuzzy-Logik
<b>Qualifikationsziele</b>	Die Studierenden kennen Grundlagen und praktische Anwendungen der Wissensverarbeitung und der Künstlichen Intelligenz. Sie können basierend auf den Kenntnissen zu ausgewählten Formen der Darstellung von Wissen und zu Problemlösungsverfahren einfache Probleme aus dem Bereich der KI analysieren und lösen.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Inhalte der Module "Modellierung" und "Algorithmen und Datenstrukturen"
<b>Literaturhinweise</b>	- I. Boersch, J. Heinsohn, R. Socher-Ambrosius: „Wissensverarbeitung. Eine Einführung in die Künstliche Intelligenz für Informatiker und Ingenieure“, Spektrum Akademischer Verlag, 2007. - W. Ertel: "Grundkurs Künstliche Intelligenz. Eine praxisorientierte Einführung", Springer 2016 - S. Russell, P. Norvig: „Künstliche Intelligenz“, Pearson, 2012. - C. Beierle, G. Kern-Isberner: „Methoden wissensbasierter Systeme“, Vieweg, 2006.
<b>Aktuelle Lehrressourcen</b>	keine

<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Pflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Anwendungsorientierte Programmierung Applied Programming
<b>Modulnummer</b>	C680 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>
<b>Dozierende</b>	Tobias Höppner
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 30 Stunden Bearbeitung Prüfungsleistung 64 Stunden Vorbereitung Lehrveranstaltung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Projektarbeit Prüfungsdauer: 4 Wochen   Wichtung: 100%
<b>Lehr- und Lernformen</b>	-
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Objektorientiertes Programmieren - Vererbung sowie Schnittstellen und Klassen als deren Implementierungen - Ausnahmebehandlung - Anwendung von generischen Datentypen, z.B. durch Arbeit mit dem Java Collection Framework - Einführung in die Gestaltung von graphischen Benutzeroberflächen
<b>Qualifikationsziele</b>	Die Studenten kennen und verstehen Syntax und Semantik der Programmiersprache Java. Sie sind in der Lage, formale und textuelle Beschreibungen von einfachen Algorithmen in kleine Programme gemäß des objektorientierten Programmierparadigmas umzusetzen sowie einfache Probleme eigenständig zu lösen. Sie kennen Grundlagen der Objektorientiertheit, können Objekte identifizieren und als Klassen implementieren.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Keine
<b>Literaturhinweise</b>	- C. Ullensboom: „Java ist auch nur eine Insel“, Galileo Computing, in der aktuellen Auflage. - F. Jobst: Programmieren in JAVA, Hanser 2015 - J. Gosling et al. : „The Java™ Language Specification“, <a href="http://docs.oracle.com/javase/specs">http://docs.oracle.com/javase/specs</a>
<b>Aktuelle Lehrressourcen</b>	Foliensatz Übungsblätter
<b>Hinweise</b>	Prüfungsvorleistung: Zwei selbstständig erarbeitete Programme (Belege). Die Abnahme und Diskussion erfolgt in einem Seminar

<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Datenbanken Database Systems
<b>Modulnummer</b>	C719 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr.-Ing. Thomas Kudraß <a href="mailto:thomas.kudrass@htwk-leipzig.de">thomas.kudrass@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr.-Ing. Thomas Kudraß <a href="mailto:thomas.kudrass@htwk-leipzig.de">thomas.kudrass@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   1 SWS Übung   1 SWS Praktikum)
<b>Selbststudienzeit</b>	94 Stunden 64 Stunden Vorbereitung Lehrveranstaltung 30 Stunden Bearbeitung Prüfungsvorleistung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Projektarbeit
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Die Teilnehmer diskutieren und trainieren auf der Grundlage der in der Vorlesung vermittelten Inhalte in Seminaren kleine Anwendungsbeispiele zum Datenbankentwurf. Grundlagen von Datenbankanfragen und Entwurfstheorie werden in Seminaren durch Übungsaufgaben vertieft. SQL als Anfragesprache wird überwiegend in Form von Rechnerübungen trainiert, ebenso die Nutzung eines relationalen Datenbanksystems. Begleitend hierzu, bearbeiten die Teilnehmer ein kleines Datenbankprojekt zu einer selbst gewählten Anwendung. Dabei durchlaufen sie drei Phasen: Analyse, Entwurf und Umsetzung (Datenbank, beispielhafte Anfragen mit Visualisierung der Ergebnisse. Dabei wird jede Projektphase vom Dozenten überprüft. Eventuelle Hinweise können die Teilnehmer somit für die nächste Projektphase bis zur endgültigen Erstellung des Projekts nutzen.
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Grundkonzepte von Datenbanken</li> <li>- Entity-Relationship-Modellierung</li> <li>- Relationales Datenmodell (Grundlagen, Relationenalgebra &amp; Relationenkalkül)</li> <li>- Logischer Datenbankentwurf (Modelltransformationen, Normalisierung)</li> <li>- Datenbanksprache SQL: Anfragen, DDL, DML</li> <li>- Integritätssicherung in Datenbanken: Constraints und Trigger</li> <li>- Transaktionen</li> <li>- Datensicherheit und Datenschutz</li> <li>- Erweiterungen relationaler Datenbanksysteme</li> <li>- praktische Übungen mit dem Datenbanksystem Oracle</li> </ul>
<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss des Moduls verfügt der Student über umfangreiche Erfahrungen bei der Nutzung von Datenbanktechnologie in einer anwendungsorientierten Sichtweise. Er kann die wichtigsten technischen Voraussetzungen beim praktischen Einsatz eines Datenbankmanagementsystems (DBMS) in einem Softwareprojekt beurteilen. Er beherrscht die Formulierung von Datenbankanfragen mittels SQL auf einem vorgegebenen Datenbankschema. Er ist in der Lage, einen Datenbankentwurf durchzuführen, ausgehend von einer Anforderungsanalyse, über die Modellierung bis hin zur Umsetzung in einem konkreten DBMS. Dabei kennt er wichtige Entwurfskriterien und kann diese bei der Modellierung der Datenbank berücksichtigen.

<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Keine
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- A. Elmasri, S. Navathe: "Grundlagen von Datenbanksystemen - Ausgabe Grundstudium", Pearson Studium, in der aktuellen Auflage.</li> <li>- A. Kemper, A. Eickler: "Datenbanksysteme", Oldenbourg, in der aktuellen Auflage.</li> <li>- T. Kudraß: "Taschenbuch Datenbanken", Hanser-Verlag, 2015.</li> <li>- K. Ramakrishnan, J. Gehrke: "Database Systems", McGraw-Hill, in der aktuellen Auflage.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistung Projekt: Datenbank-Projekt (2 Belege und Praktikum)
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Pflichtmodul Medieninformatik   Bachelor (20MIB) Pflichtmodul Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Digitaltechnik I Digital Electronics I
<b>Modulnummer</b>	C752 Version: 1
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Hanna Brodowsky <a href="mailto:hanna.brodowsky@htwk-leipzig.de">hanna.brodowsky@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	8 ECTS-Punkte
<b>Workload</b>	240 Stunden
<b>Lehrveranstaltungen</b>	5 SWS (3 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	170 Stunden 50 Stunden Bearbeitung Prüfungsvorleistung 50 Stunden Selbststudium 40 Stunden E-Learning 30 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Bearbeiten von Problemen und Lösungsfindung, Einzel- und Gruppenarbeit
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Signale - Zahlsysteme, Informationsdarstellung und Codierung - Schaltalgebra - Schaltungstechnik - Synthese und Analyse von Schaltnetzen - Realisierung spezieller Schaltnetze
<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, schaltalgebraische Beschreibungsmethoden für unterschiedliche technische Aufgabenstellungen anzuwenden. Sie können durch ihr Wissen mittels verschiedener Methoden und Verfahren Schaltnetze selbsttätig entwerfen, optimieren und technisch umsetzen.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Fähigkeit zum logischen und algorithmischen Denken. Geübter Umgang mit den physikalischen Grundgrößen und ihren Maßeinheiten sowie ihre Anwendung auf Gleichstromkreise. Aus verbalen Aufgabenstellungen heraus können Gleichungen und Gleichungssysteme aufgestellt und mit den Methoden der Arithmetik gelöst werden.
<b>Literaturhinweise</b>	- K. Fricke: „Digitaltechnik“, Vieweg, in der aktuellen Auflage. - G. Wöstenkühler: „Grundlagen der Digitaltechnik“, Hanser, in der aktuellen Auflage. - G. Scarbata: „Synthese und Analyse Digitaler Schaltungen“, Oldenbourg, in der aktuellen Auflage.

<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Belege: Es werden 4 Belege ausgereicht. Dabei müssen mindestens 50% der Punkte der Gesamtbelegleistung erreicht werden.
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Pflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	IT-Sicherheit IT Security
<b>Modulnummer</b>	C799 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Karsten Weicker <a href="mailto:karsten.weicker@htwk-leipzig.de">karsten.weicker@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Karsten Weicker <a href="mailto:karsten.weicker@htwk-leipzig.de">karsten.weicker@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Präsentation
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Klassische Vorlesungen werden durch verschiedene Formate im Übungsbetrieb ergänzt. Dort kommen zum Einsatz: vorab zuhause zu bearbeitende Übungsaufgaben, praktische Aufgaben am Computer (z.B. Angriff durch Command-Injection, Entschlüsselung einer Vigenère-Chiffre, partielle Entschlüsselung von Mini-AES durch eine Square-Attacke), Simulationen durch das Kartenspiel Elevation of Privileges und studentische Vorträge in den letzten vier Wochen.
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Grundbegriffe, Bedrohungen, Risiken</li> <li>- Analyse von Bedrohungen und Risiken</li> <li>- Zugriffskontrollmodelle und deren Anwendungen</li> <li>- verwundbare und sichere Sprachkonstrukte in der Anwendungsprogrammierung</li> <li>- Umsetzung von Sicherheitskonzepten mit Mitteln der Hard- und Software</li> <li>- Informationssicherheit in Unternehmen: ISO 27001 und Grundsatz</li> <li>- Verschlüsselung, Authentifizierung und Zertifizierung mit symmetrischen und asymmetrischen kryptographischen Bausteinen</li> <li>- Sensibilisierung zu Passwörtern, Social Engineering und Phishing</li> <li>- Praktische Übungen zur Realisierung von Maßnahmen der Sicherheit</li> </ul>
<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss der Lehrveranstaltung sind die Studierenden in der Lage, Bedrohungen von Rechnern und Netzen zu erkennen und den Schutzbedarf dieser Ressourcen einzuschätzen. Sie haben ein Grundverständnis für Verwundbarkeiten und Risiken und kennen die gängigen Sicherheitsmechanismen. Sie können Sicherheitsprinzipien, -mechanismen und -vorkehrungen bei der Konfiguration und Implementierung von Softwarelösungen anwenden.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Die Studierenden sind sowohl mit den Wirkprinzipien von Rechnern, der Anwendungsprogrammierung, der Rolle und Funktionsweise von Betriebssystemen sowie mit der Kommunikation von Rechnern über Netze vertraut.

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- A. Shostack: "Threat Modeling: Designing for Security", Wiley, 2014</li> <li>- R. J. Anderson: "Security Engineering", Wiley, 2010.</li> <li>- C. Eckert. : "IT-Sicherheit", Oldenburg, 2008.</li> <li>- M. Howard, D. LeBlanc, J. Viega: "24 Deadly Sins of Software Security", Mc Graw Hill, 2010.</li> <li>- H. Kersten, J. Reuter, K.-W. Schröder: "IT-Sicherheitsmanagement nach ISO 27001 und Grundsatz", SpringerVieweg, 2013.</li> <li>- <a href="https://www.bsi.bund.de/">https://www.bsi.bund.de/</a></li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Rechnerarchitektur Computer Architecture
<b>Modulnummer</b>	C827 Version: 0
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Dr. rer. nat. Sebastian Rinke <a href="mailto:sebastian.rinke@htwk-leipzig.de">sebastian.rinke@htwk-leipzig.de</a>
<b>Dozierende</b>	Dr. rer. nat. Sebastian Rinke <a href="mailto:sebastian.rinke@htwk-leipzig.de">sebastian.rinke@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Referat
<b>Prüfungsleistung(en)</b>	Prüfung mündliches Fachgespräch Prüfungsdauer: 30 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Bearbeiten von Problemen und Lösungsfindung, Präsentationen
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Grundlagen der Rechnerarchitektur</li> <li>- Prozessortypen und Befehlssätze</li> <li>- Leistungsbewertung</li> <li>- Pipelineverarbeitung</li> <li>- Speichersysteme und Speicherverwaltung</li> <li>- Konzepte der Parallelverarbeitung und parallele Rechnerarchitekturen</li> </ul>
<b>Qualifikationsziele</b>	Die Studierenden sollen strukturelle, organisatorische und implementierungstechnische Aspekte verschiedener Rechnerarchitekturen interpretieren können. Des Weiteren sollen sie in die Lage versetzt werden, die Leistung derartiger Systeme bewerten zu können, wozu sie verschiedene Verfahren und Methoden anwenden. Ein besonderes Augenmerk liegt auf den Möglichkeiten der Parallelarbeit und den damit verbundenen Rechnerarchitekturvarianten, die hinsichtlich ihres Einsatzspektrums sowie der Vor- und Nachteile eingeordnet werden können.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Verständnis zu allgemeinen Hardwaregrundlagen insbesondere zu digitalen Schaltungen
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- R. Hellmann: „Rechnerarchitektur“, Oldenbourg, in der aktuellen Auflage.</li> <li>- A. Böttcher: „Rechneraufbau und Rechnerarchitektur“, Springer, in der aktuellen Auflage.</li> <li>- A. S. Tanenbaum: „Computerarchitektur“, Pearson Studium, in der aktuellen Auflage.</li> <li>- T. Rauber, G. Rünger: „Parallele Programmierung“, Springer, in der aktuellen Auflage.</li> <li>- H. G. Kruse: „Leistungsbewertung bei Computersystemen“, Springer, in der aktuellen Auflage.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Pflichtmodul

Link zu Kurs/Lernressourcen im  
OPAL/Moodle/etc.

<b>Modul</b>	Wissenschaftskommunikation in der Informatik Writing and Presenting in Computer Science
<b>Modulnummer</b>	C902 Version: 1
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommer- und Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Thomas Riechert <a href="mailto:thomas.riechert@htwk-leipzig.de">thomas.riechert@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Thomas Riechert <a href="mailto:thomas.riechert@htwk-leipzig.de">thomas.riechert@htwk-leipzig.de</a> Dozentin/Dozent in: "Wissenschaftskommunikation"  Dr. Antje Tober-Nietner <a href="mailto:antje.tober@htwk-leipzig.de">antje.tober@htwk-leipzig.de</a> Dozentin/Dozent in: "Technisches Englisch"
<b>Sprache(n)</b>	Deutsch in "Wissenschaftskommunikation"  Englisch in "Technisches Englisch"
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden 90 Stunden in "Wissenschaftskommunikation" 60 Stunden in "Technisches Englisch"
<b>Lehrveranstaltungen</b>	4 SWS (1 SWS Vorlesung   3 SWS Seminar) 2 SWS (1 SWS Vorlesung   1 SWS Seminar) in "Wissenschaftskommunikation" 2 SWS (2 SWS Seminar) in "Technisches Englisch"
<b>Selbststudienzeit</b>	94 Stunden 62 Stunden in "Wissenschaftskommunikation" 32 Stunden in "Technisches Englisch"
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Projektarbeit Prüfungsdauer: 10 Wochen   Wichtung: 60%   nicht kompensierbar in "Wissenschaftskommunikation"  Prüfung Referat Modulprüfung   Prüfungsdauer: 15 Minuten   Wichtung: 40%   nicht kompensierbar
<b>Lehr- und Lernformen</b>	<b>Wissenschaftskommunikation:</b> Der fachliche Teil orientiert sich im Ablauf an der Erstellung, Einreichung, Begutachtung und Überarbeitung eines wissenschaftlichen Papers bei einer Fachtagung.  <b>Technisches Englisch:</b> Typische Merkmale der Informatikfachsprache Englisch werden vorgestellt und geübt. Im zweiten Teil der Lehrveranstaltung wird eine englischsprachige Präsentation zur erstellten Hausarbeit erwartet.
<b>Medienform</b>	<b>Wissenschaftskommunikation:</b> keine Angabe  <b>Technisches Englisch:</b> keine Angabe

<b>Lehrinhalte/Gliederung</b>	<p><b>Wissenschaftskommunikation:</b></p> <ul style="list-style-type: none"> <li>- Literaturrecherche, Informatik als Wissenschaft, wissenschaftliches Schreiben, Einführung in Latex, Begutachtung wissenschaftlicher Arbeiten, Wissenschaftsethik, wissenschaftliche Vorträge</li> <li>- Erarbeitung und gegenseitige Begutachtung einer eigenen Arbeit entsprechend der typischen Organisation einer wissenschaftlichen Tagung</li> </ul> <p><b>Technisches Englisch:</b></p> <ul style="list-style-type: none"> <li>- Präsentation einer eigenen Arbeit</li> <li>- Englisch als Fachsprache der Informatik: numbers, mathematical symbols and operations, databases, complex systems, programming, spreadsheets, product lifecycle management, electronic learning, licenses</li> </ul>
<b>Qualifikationsziele</b>	<p>Die Studierenden können nach Abschluss des Moduls mit englischsprachiger Fachliteratur umgehen, eigene Ergebnisse und Ausführungen gemäß den in der Informatik üblichen Konventionen verschriftlichen, Paper anderer Autoren begutachten und ihre eigenen Ergebnisse in Form einer englischsprachigen Präsentation halten.</p> <p>Nach erfolgreichem Abschluss des Moduls haben sich die Studierenden ausgewählte Teilbereiche ihres Studienfachs in der Fremdsprache angeeignet und sind in der Lage, die englische Fachsprache in diversen studien- und berufsbezogenen Kontexten sowohl mündlich (als auch schriftlich) sicher anzuwenden. Die Studierenden können:</p> <ul style="list-style-type: none"> <li>- längere Redebeiträge und Vorträge im Fach verstehen und auch komplexer Argumentation folgen,</li> <li>- spontan und fließend über konkrete Fachgegenstände sprechen und deren Facetten diskutieren,</li> <li>- komplexe Sachtexte und Fachartikel sowie technische Anleitungen verstehen.</li> </ul>
<b>Zulassungsvoraussetzung</b>	Erfolgreicher Abschluss des Moduls "Englisch für Studium und Beruf (B2)"
<b>Empfohlene Voraussetzungen</b>	Sprachkenntnisse auf Niveau B2 des Gemeinsamen europäischen Referenzrahmens (GER).
<b>Literaturhinweise</b>	<p><b>Wissenschaftskommunikation:</b></p> <ul style="list-style-type: none"> <li>- H. Balzert et al.: "Wissenschaftliches Arbeiten - Wissenschaft, Quellen, Artefakte, Organisation, Präsentation" W3L, in der aktuellen Auflage.</li> <li>- J. Zobel: "Writing for Computer Science", 2. Auflage, Springer, 2004.</li> <li>- L. Dupré: "BUGS in Writing", Addison-Wesley, 1998.</li> <li>- S. Peyton Jones et al.: "How to give a good research talk", SIGPLAN Notices 28(11), S. 9-12, 1993.</li> <li>- I. Parberry: "A guide for new referees in theoretical computer science", Information and Computation, 112(1):96-116, 1994.</li> <li>- G. Cormode: "How NOT to review a paper: The tools and techniques of the adversarial reviewer", SIGMOD Record, 37(4), S. 100-104, 2008.</li> </ul> <p><b>Technisches Englisch:</b> Zusatz- und Übungsmaterial (PC, Audio, Video, Print) im Sprachlernzentrum (SLZ) verfügbar. Eine aktuelle Literaturempfehlung erfolgt zu Semesterbeginn durch den/die Dozenten/in.</p>
<b>Aktuelle Lehrressourcen</b>	<p><b>Wissenschaftskommunikation:</b> keine</p> <p><b>Technisches Englisch:</b> keine</p>
<b>Hinweise</b>	<p><b>Wissenschaftskommunikation:</b> 62h Bearbeitung der Prüfungsleistung: Projektleistung in Form der Erstellung der wissenschaftlichen Hausarbeit und Begutachtung anderer Hausarbeiten</p>
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Digitaltechnik II Digital Electronics II
<b>Modulnummer</b>	C947 Version: 1
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Dr. rer. nat. Sebastian Rinke <a href="mailto:sebastian.rinke@htwk-leipzig.de">sebastian.rinke@htwk-leipzig.de</a>
<b>Dozierende</b>	Dr. rer. nat. Sebastian Rinke <a href="mailto:sebastian.rinke@htwk-leipzig.de">sebastian.rinke@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Bearbeiten von Problemen und Lösungsfindung, Einzel- und Gruppenarbeit
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Theoretische Grundlagen der Schaltwerke</li> <li>- Synthese von Schaltwerken</li> <li>- Analyse von Schaltwerken</li> <li>- Realisierung spezieller Schaltwerke</li> <li>- Grundlagen der Informations- und Codierungstheorie, Datenkompression und Codesicherung</li> </ul>
<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, einerseits beliebige Schaltwerke bis zu einem bestimmten Komplexitätsgrad zu entwerfen und zu analysieren und andererseits die wichtigsten Standard-Schaltwerke hinsichtlich ihrer Funktionsweise zu interpretieren. Für verschiedene Aufgabenstellungen können grundlegende Datenkompressions- und Codesicherungsverfahren angewandt werden. Zusammenhänge zu angrenzenden Gebieten der Informatik werden dabei verdeutlicht und führen zu vertieften Kenntnissen über informationsverarbeitende Systeme aus Sicht der Hardware.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Theoretische und physikalische Grundlagen der Informatik, Fähigkeit zum Entwurf von Schaltnetzen
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- K. Fricke: „Digitaltechnik“, Vieweg, in der aktuellen Auflage.</li> <li>- G. Wöstenkühler: „Grundlagen der Digitaltechnik“, Hanser, in der aktuellen Auflage.</li> <li>- G. Scarbata: „Synthese und Analyse Digitaler Schaltungen“, Oldenbourg, in der aktuellen Auflage.</li> <li>- W. Dankmeier: „Codierung“, Vieweg, in der aktuellen Auflage.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistung Belege: Es werden 4 Belege ausgereicht. Dabei müssen mindestens 50% der Punkte der Gesamtbelegleistung erreicht werden.
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Pflichtmodul

Link zu Kurs/Lernressourcen im  
OPAL/Moodle/etc.

<b>Modul</b>	Grundlagen der Programmierung Introduction to Programming
<b>Modulnummer</b>	C963 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	7 ECTS-Punkte
<b>Workload</b>	210 Stunden
<b>Lehrveranstaltungen</b>	4.50 SWS (2.50 SWS Vorlesung   2 SWS Praktikum)
<b>Selbststudienzeit</b>	147 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung mündliches Fachgespräch Prüfungsdauer: 20 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Programmierübungen in den Seminaren - Einsatz von e-Learning mit automatisierten Tests zur Eigenkontrolle
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Einführung in die Programmierung und Hardware - Begriff des Algorithmus - Programmablaufpläne und Struktogramme - Imperative Programmierung - Kontrollstrukturen - Unterprogramme  - Objektorientiertes Programmieren - Verwendung von objektorientierten Datenstrukturen - Vererbung sowie Schnittstellen und Klassen als deren Implementierungen - Ausnahmebehandlung - Vererbung  - Grundlagen des Umgangs mit Dateien und Speicher
<b>Qualifikationsziele</b>	Die Studenten kennen und verstehen Syntax und Semantik der Programmiersprache C++. Sie sind in der Lage, formale und textuelle Beschreibungen von einfachen Algorithmen in kleine Programme gemäß des imperativen und objektorientierten Programmierparadigmas umzusetzen sowie einfache Probleme eigenständig zu lösen. Sie kennen Grundlagen der Objektorientiertheit, können Objekte identifizieren und als Klassen implementieren.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Keine
<b>Literaturhinweise</b>	- U. Breymann: „Der C++ Programmierer“, Hanser, 2015. - B. Stroustrup: „Die C++ Programmiersprache“, Hanser, 2015.
<b>Aktuelle Lehrressourcen</b>	keine

<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Automaten und formale Sprachen Automata and Formal Languages
<b>Modulnummer</b>	C993 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 14 Stunden Vorbereitung Lehrveranstaltung 70 Stunden Bearbeitung Prüfungsvorleistung 10 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg Prüfungsvorleistung Präsentation
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Vorlesung - Übung - Bearbeiten von Problemen und Lösungsfindung, - Selbststudium anhand theoretischer und praktischer Übungsaufgaben
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	Formale Sprachen und verschiedene Darstellungsformen dafür, reguläre Ausdrücke Grammatiken (Chomsky-Hierarchie, Pumping Lemmata)  Berechnungsmodelle: endliche Automaten, Kellerautomaten, Turingmaschinen  Ausblick auf Grenzen der Berechenbarkeit
<b>Qualifikationsziele</b>	Die Studierenden sind in der Lage, wichtige Klassen formaler Sprachen als Grundlage von Programmier- und Beschreibungssprachen einzuordnen und kennen die wesentlichen Eigenschaften der Sprachklassen. Sie kennen die entsprechenden abstrakten Maschinenmodelle und Algorithmen und können sie zur Darstellung und Lösung praktischer Aufgabenstellungen einsetzen. Die Studierenden wissen, dass nicht jedes formal darstellbare Problem algorithmisch lösbar ist.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	anwendungsbereite Kenntnisse auf den Gebieten Modellierung, Logik und Programmierung

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- J. E. Hopcroft, J. D. Ullman: "Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie", Addison-Wesley, aktuelle Auflage.</li> <li>- U. Schöning: "Theoretische Informatik - kurzgefasst", Spektrum, aktuelle Auflage.</li> <li>- D. Hoffmann: "Theoretische Informatik", Hanser, 2009.</li> <li>- R. Socher: "Theoretische Grundlagen der Informatik", Hanser, 2008</li> <li>- G. Vossen, K.-U. Witt: "Grundkurs Theoretische Informatik", Springer Vieweg, aktuelle Auflage.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistung: regelmäßiges erfolgreiches Lösen der praktischen Übungsaufgaben und 3 Kurzvorträge zu schriftlichen Übungsaufgaben
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Wahlpflicht</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Englisch für Studium und Beruf (B2) Academic and Vocational English (B2)
<b>Modulnummer</b>	F742 Version: 2
<b>Fakultät</b>	HSK-FI: Hochschulkolleg - Fremdsprachen und Interkulturalität
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommer- und Wintersemester
<b>Modulverantwortliche</b>	Dr. Antje Tober-Nietner <a href="mailto:antje.tober@htwk-leipzig.de">antje.tober@htwk-leipzig.de</a>
<b>Dozierende</b>	Ipsita Kloss <a href="mailto:ipsita.kloss@htwk-leipzig.de">ipsita.kloss@htwk-leipzig.de</a> Dozentin/Dozent in: "Englisch für Studium und Beruf (B2)"  Sanaz Bakhshi Nia <a href="mailto:sanaz.bakhshi_nia@htwk-leipzig.de">sanaz.bakhshi_nia@htwk-leipzig.de</a> Dozentin/Dozent in: "Englisch für Studium und Beruf (B2)"  Gwen Davies-Schneider <a href="mailto:gwen.davies_schneider@htwk-leipzig.de">gwen.davies_schneider@htwk-leipzig.de</a> Dozentin/Dozent in: "Englisch für Studium und Beruf (B2)"  M. A. EB Dietlind Unger <a href="mailto:dietlind.unger@htwk-leipzig.de">dietlind.unger@htwk-leipzig.de</a> Dozentin/Dozent in: "Englisch für Studium und Beruf (B2)"  Dr. Antje Tober-Nietner <a href="mailto:antje.tober@htwk-leipzig.de">antje.tober@htwk-leipzig.de</a> Dozentin/Dozent in: "eXplore English Terms"
<b>Sprache(n)</b>	Englisch in "Englisch für Studium und Beruf (B2)"  Englisch in "eXplore English Terms"
<b>ECTS-Leistungspunkte</b>	3 ECTS-Punkte
<b>Workload</b>	90 Stunden 60 Stunden in "Englisch für Studium und Beruf (B2)" 30 Stunden in "eXplore English Terms"
<b>Lehrveranstaltungen</b>	2 SWS (2 SWS Seminar) 2 SWS (2 SWS Seminar) in "Englisch für Studium und Beruf (B2)" 0 SWS in "eXplore English Terms"
<b>Selbststudienzeit</b>	90 Stunden 60 Stunden in "Englisch für Studium und Beruf (B2)" 30 Stunden in "eXplore English Terms"
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung am Computer in "eXplore English Terms"
<b>Prüfungsleistung(en)</b>	Prüfung Referat Prüfungsdauer: 15 Minuten   Wichtung: 25%   nicht kompensierbar in "Englisch für Studium und Beruf (B2)"  Prüfung Klausurarbeit Prüfungsdauer: 90 Minuten   Wichtung: 75%   nicht kompensierbar in "Englisch für Studium und Beruf (B2)"

<b>Lehr- und Lernformen</b>	<p><b>Englisch für Studium und Beruf (B2):</b> Seminar</p> <p><b>eExplore English Terms:</b> Terminologietrainer</p>
<b>Medienform</b>	<p><b>Englisch für Studium und Beruf (B2):</b></p> <ul style="list-style-type: none"> <li>- Präsentationen</li> <li>- Übungsblätter</li> <li>- Tafelbild</li> <li>- Lehrfilme</li> <li>- Hörbeispiele</li> </ul> <p><b>eExplore English Terms:</b> E-Learning-Modul</p>
<b>Lehrinhalte/Gliederung</b>	<p><b>Englisch für Studium und Beruf (B2):</b></p> <ul style="list-style-type: none"> <li>- mündliche Kommunikation in Studium und Beruf (z. B. Fachvorträge, Präsentationen, Diskussionen),</li> <li>- schriftliche Kommunikation in Studium und Beruf (z. B. E-Mails, Lebenslauf, Bewerbungen),</li> <li>- Sprachstrukturen, Grammatik und Terminologie für Studium und Beruf.</li> </ul> <p><b>eExplore English Terms:</b> Learn 200 terms in your field of study by doing more than 1,000 practice exercises.</p>
<b>Qualifikationsziele</b>	<p>Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage:</p> <ul style="list-style-type: none"> <li>- komplexe studien- und berufsrelevante Hör- und Lesetexte, auch zu weniger vertrauten Themen, zu verstehen,</li> <li>- unter Verwendung vielfältiger, auch komplexer sprachlicher Mittel studien- und berufsrelevante Texte aus bekannten Themenbereichen zu verfassen,</li> <li>- unter Verwendung vielfältiger, auch komplexer sprachlicher Mittel studien- und berufsrelevante Gesprächssituationen, in denen es um komplexe Themen aus bekannten Themenbereichen geht, sicher zu bewältigen,</li> <li>- Sachverhalte ausführlich zu erläutern und Standpunkte zu verteidigen.</li> </ul>
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Sprachkenntnisse auf mittlerem Niveau bzw. entsprechendes Ergebnis im Einstufungstest des Bereichs Fremdsprachen und Interkulturalität im Hochschulkolleg.
<b>Literaturhinweise</b>	<p><b>Englisch für Studium und Beruf (B2):</b> Zusatz- und Übungsmaterial (PC, Audio, Video, Print) im Sprachlernzentrum (SLZ) verfügbar. Eine aktuelle Literaturempfehlung erfolgt zu Semesterbeginn durch den/die Dozenten/in.</p> <p><b>eExplore English Terms:</b> keine Angabe</p>
<b>Aktuelle Lehrressourcen</b>	<p><b>Englisch für Studium und Beruf (B2):</b> Keine</p> <p><b>eExplore English Terms:</b> keine</p>
<b>Hinweise</b>	<p>Die Fremdsprachen Französisch, Spanisch und Russisch werden jährlich ab dem Sommersemester angeboten und sind zweisemestrige Kurse. Bei Interesse statt Englisch eine andere Fremdsprache zu wählen, tragen Sie sich bitte hier bereits im 1. Semester Ihres Studiums ein:  <a href="https://bildungspotrait.sachsen.de/opal/auth/RepositoryEntry/444465162/CourseNode/102602284957609">https://bildungspotrait.sachsen.de/opal/auth/RepositoryEntry/444465162/CourseNode/102602284957609</a>.</p>
<b>Verwendbarkeit</b>	Das Modul ist in Bachelorstudiengängen verwendbar.
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Mathematik für Informatiker II Mathematics in Computer Science II
<b>Modulnummer</b>	N055 Version: 3
<b>Fakultät</b>	MNZ-Ma: Mathematik - Mathematisch-Naturwissenschaftliches Zentrum
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. habil. Martin Grüttmüller <a href="mailto:martin.gruettmueller@htwk-leipzig.de">martin.gruettmueller@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. habil. Martin Grüttmüller <a href="mailto:martin.gruettmueller@htwk-leipzig.de">martin.gruettmueller@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung am Computer Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Vorlesung: Tafel und Beamer, ein Lückenskript wird bereitgestellt  Seminare: Vertiefung der Vorlesungsinhalte durch eigenständiges Lösen von Aufgaben
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	Norm, Skalarprodukt, Determinanten, Bestimmtes und unbestimmtes Integral, Integrationsmethoden, Hauptsatz der Differenzial- und Integralrechnung, uneigentliches Integral, Fourier-Reihen, Funktionen mehrerer Veränderlicher, Kurven, partielle Ableitungen, Gebietsintegral, Substitution des Gebietsintegrals, Definitheit von Matrizen und Extrema bei Funktionen mehrerer Veränderlicher.
<b>Qualifikationsziele</b>	Mit der Einführung der Determinanten und Eigenwerte verfügen die Studierenden über weitere Möglichkeiten zur Charakterisierung von Matrizen und linearen Abbildungen. Mit der Betrachtung von Potenzreihen lernen Studierende Darstellungsmöglichkeiten elementarer Funktionen und Möglichkeiten zur deren Darstellung auf Rechnern kennen. Der Begriff des bestimmten Integrals wird geometrisch motiviert; die Verbindung zwischen Integral- zur Differenzialrechnung wird aufgezeigt.  Die Studierenden beherrschen die wichtigsten Methoden zur Bestimmung bestimmter und unbestimmter Integrale. Im Rahmen der Integralrechnung werden auch uneigentliche Integrale behandelt. Mit der Fourier-Analyse lernen Studierende ein wichtiges Anwendungsbereich der Integralrechnung kennen. Mit der Übergang der Grundbegriffe (Konvergenz, Stetigkeit, Ableitung, Integral) auf Funktionen mehrerer Veränderlicher und exemplarischen Anwendungen erwerben die Studierenden ein tieferes Verständnis für das Zusammenspiel mathematischer Methoden aus Analysis und Algebra in der Informatik.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Mathematik für Informatiker I

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- Konkrete Mathematik (nicht nur) für Informatiker, Edmund Weitz, Springer Spektrum 2018</li> <li>- Höhere Mathematik in Rezepten, 2. Auflage, Christian Karpfinger, Springer Spektrum 2015</li> <li>- Arbeitsbuch Höhere Mathematik in Rezepten, Christian Karpfinger, Springer Spektrum 2014</li> <li>- Mathematik für Informatiker, Steffen Goebbel, Jochen Rethmann, Springer Vieweg 2014</li> <li>- Mathematik für Informatiker, 2. Auflage, Matthias Schubert, Springer Vieweg+Teubner 2012</li> <li>- Mathematik für Informatiker, Band 1, 4. Auflage, Gerald Teschl, Susanne Teschl, Springer Vieweg 2013</li> <li>- Mathematik für Informatiker, Band 2, 3. Auflage, Gerald Teschl, Susanne Teschl, Springer Vieweg 2014</li> <li>- Mathematik für Informatiker, 2. Auflage, Dirk Hachenberger, Pearson Studium 2008</li> <li>- Toolbox Mathematik für MINT-Studiengänge, Erhard Cramer, Udo Kamps, Jessica Lehmann, Sebastian Walcher, Springer Spektrum 2017</li> <li>- So einfach ist Mathematik – Zwölf Herausforderungen im ersten Semester, Dirk Langemann, Vanessa Sommer, Springer Spektrum 2017</li> <li>- Mathematik-Klausurtrainer, Reinhard Strehlow, Hanser 2007.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Mathematik für Informatiker III Mathematics in Computer Science III
<b>Modulnummer</b>	N417 Version: 0
<b>Fakultät</b>	MNZ-Ma: Mathematik - Mathematisch-Naturwissenschaftliches Zentrum
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. habil. Andreas Lasarow <a href="mailto:andreas.lasarow@htwk-leipzig.de">andreas.lasarow@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. habil. Andreas Lasarow <a href="mailto:andreas.lasarow@htwk-leipzig.de">andreas.lasarow@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 42 Stunden Vorbereitung Lehrveranstaltung 20 Stunden Bearbeitung Prüfungsvorleistung 10 Stunden Selbststudium 22 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Vorlesungen: Tafel und Beamer, wobei Folien via OPAL bereitgestellt werden - Seminare: Tafel, Lösen von Übungsaufgaben.
<b>Medienform</b>	Tafel und Beamer
<b>Lehrinhalte/Gliederung</b>	- Beschreibende Statistik: Häufigkeitsverteilungen, Histogramme, Box-Plots, Lagemaße, Streuungsmaße  - Wahrscheinlichkeitsrechnung: Wahrscheinlichkeitsräume, bedingte Wahrscheinlichkeiten, Satz von Bayes, stochastische Unabhängigkeit von Ereignissen, Zufallsgrößen, Verteilungsfunktionen, Grenzwertsätze  - Induktive Statistik: Parameterschätzungen, Maximum-Likelihood-Methode, Konfidenzintervalle, Grundlegendes zu Hypothesentests am Beispiel eines t-Tests
<b>Qualifikationsziele</b>	Das Hauptaugenmerk liegt in der Vermittlung wahrscheinlichkeitstheoretischer Grundlagen sowie prinzipieller Vorgehensweisen hinsichtlich einer statistischen Auswertung konkreter Stichproben. Insbesondere soll die Fähigkeit erworben werden, Verfahren bei der Untersuchung zufallsabhängiger Phänomene sachgerecht einzusetzen.  Nach erfolgreichem Abschluss beherrscht der Studierende wahrscheinlichkeitstheoretische Grundbegriffe und Denkweisen sowie elementare Werkzeuge der statistischen Analyse gegebener Daten. Hierdurch wird er insbesondere in die Lage versetzt, weitere Kenntnisse auf dem Gebiet der Stochastik zu erwerben, die es ermöglichen, praktische Probleme zu lösen.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Mathematik für Informatiker I, II

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- Beyer, O./, Erfurth, H.: Wahrscheinlichkeitsrechnung und mathematische Statistik, Teubner, 1999</li> <li>- Bosch, K.: Elementare Einführung in die Wahrscheinlichkeitsrechnung, Vieweg+Teubner, 2011.</li> <li>- Bosch, K.: Statistik für Nichtstatistiker - Zufall und Wahrscheinlichkeit, Oldenbourg Wissenschaftsverlag, 2012</li> <li>- Cramer, E./Kamps, U.: Grundlagen der Wahrscheinlichkeitsrechnung und Statistik - Eine Einführung für Studierende der Informatik, der Ingenieur- und Wirtschaftswissenschaften, Springer, 2017</li> <li>- Hübner, G.: <b>Stochastik - Eine anwendungsorientierte Einführung für Informatiker, Ingenieure und Mathematiker</b>, Vieweg+Teubner, 2009</li> <li>- Storm, R.: Wahrscheinlichkeitsrechnung, mathematische Statistik und statistische Qualitätskontrolle, Leipziger Fachbuchverlag, 2007</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Pflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Mathematik für Informatiker I Mathematics in Computer Science I
<b>Modulnummer</b>	N662 Version: 3
<b>Fakultät</b>	MNZ-Ma: Mathematik - Mathematisch-Naturwissenschaftliches Zentrum
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. habil. Martin Grüttmüller <a href="mailto:martin.gruettmueller@htwk-leipzig.de">martin.gruettmueller@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. habil. Martin Grüttmüller <a href="mailto:martin.gruettmueller@htwk-leipzig.de">martin.gruettmueller@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	7 ECTS-Punkte
<b>Workload</b>	210 Stunden
<b>Lehrveranstaltungen</b>	6 SWS (4 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	126 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung am Computer Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Vorlesung: Tafel und Beamer, ein Lückenskript wird bereitgestellt - Seminare: Vertiefung der Vorlesungsinhalte durch eigenständiges Lösen von Aufgaben
<b>Medienform</b>	Tafel und Beamer
<b>Lehrinhalte/Gliederung</b>	Mengen, Aussagen, Beweistechniken, Algebraische Strukturen, Vektorräume, Basis und Dimension, Lineare Abbildungen und Matrizen, Lineare Gleichungssysteme. Ungleichungen, Folgen und Konvergenz, Stetigkeit, Grenzwertsätze, Reihen, Ableitung und Anwendungen der Differenzialrechnung
<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss des Moduls kennen die Studierenden die wichtigsten mathematischen Konzepte, welche für die Informatik von Bedeutung sind. Hierzu gehört ein solides mathematisches Grundwissen über Mengen, Aufbau des Zahlensystems, Aussagen, Abbildungen und grundlegende Beweistechniken. Im Bereich der Algebra kennen die Studierenden die Vektorraumstruktur und wissen die geometrischen, arithmetischen sowie strukturbetont-abstrakten Aspekte Informatik-bezogen einsetzen. Die Studierenden beherrschen alle Gesichtspunkte der Vektorräume, wozu der sichere Umgang mit den zentralen Begriffen - Lineare Abhängigkeit/Unabhängigkeit, Basis, Dimension, Teilraumstrukturen und Lineare Abbildungen - zählt. Die Studierenden lernen mit Linearen Gleichungssystemen eine der wichtigsten Aufgaben der linearen Algebra kennen und eignen sich fundierte Kenntnisse zu deren Lösung und deren Einordnung in den Gesamtkomplex der Linearen Algebra an. Ferner haben die Studierenden ein tiefes Verständnis für den Zusammenhang zwischen Matrizen und linearen Abbildungen entwickelt. Im Bereich der Analysis lernen die Studierenden den Umgang mit Ungleichungen und Abschätzungen. Grundlage der Analysis ist das Beherrschung von Folgen und deren Konvergenzverhalten. Mit deren Anwendung im Rahmen der Analyse von Algorithmen werden Bezüge zur Informatik aufgezeigt. Mit Reihen lernen Studierende weitere (spezielle) Folgen kennen. Neben der Stetigkeit von Funktionen einer Veränderlichen wird das Studium elementarer Funktionen und deren Eigenschaften vermittelt. Mit der Ableitung und den wichtigsten Ableitungsregeln lernen die Studierenden ein wichtiges Werkzeug zur Untersuchung des Verhaltens von Funktionen kennen. Im Rahmen der Differenzialrechnung lernen die Studierenden Bedingungen für Extrema, die Regeln von de l'Hospital und die Approximation von Funktionen durch Taylor-Polynome kennen.
<b>Zulassungsvoraussetzung</b>	Keine

<b>Empfohlene Voraussetzungen</b>	Keine
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- O. Bretscher: "Linear Algebra with Applications", Pearson, in der aktuellen Auflage.</li> <li>- M. Brill: "Mathematik für Informatiker", Hanser, 2005, 2. Auflage</li> <li>- H.-J. Dobner, G. Dobner: "Lineare Algebra", Spektrum, in der aktuellen Auflage.</li> <li>- H.-J. Dobner, B. Engelmann: "Analysis I", Spektrum, in der aktuellen Auflage.</li> <li>- D. Hachenberger: "Mathematik für Informatiker", Pearson, 2008.</li> <li>- B. Thomas, M. D. Weir: "Analysis 1", Pearson, 2014, 12. Auflage.</li> <li>- H. D: Vinod: "Hands_On Matrix Algebra Using R", World Scientific, 2011.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Pflichtmodul</p> <p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Studium generale General Studies
<b>Modulnummer</b>	U622 Version: 0
<b>Fakultät</b>	HSK-SG: Hochschulkolleg - Studium generale
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommer- und Wintersemester
<b>Modulverantwortliche</b>	Dr. rer. nat. Martin Schubert <a href="mailto:martin.schubert@htwk-leipzig.de">martin.schubert@htwk-leipzig.de</a>
<b>Dozierende</b>	
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	2 ECTS-Punkte
<b>Workload</b>	60 Stunden
<b>Lehrveranstaltungen</b>	2 SWS (2 SWS Vorlesung)
<b>Selbststudienzeit</b>	32 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Teilnahmebescheinigung Wichtung: 100%   nicht benotet
<b>Lehr- und Lernformen</b>	keine Angabe
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	Im Studium generale werden gesellschaftsrelevante Themen und wissenschaftlich/technologische Fragestellungen mit fachübergreifendem Charakter behandelt. Dabei soll der Blick auf die Funktions- und Kommunikationsmechanismen in unserer Gesellschaft geschärft werden. Die Bearbeitung eines Themas erfolgt aus möglichst unterschiedlichen Perspektiven.  Zur Realisierung des Lernziels werden Lehrveranstaltungen mit unterschiedlichen Lehrinhalten angeboten, aus denen je nach Platzangebot frei gewählt werden kann.
<b>Qualifikationsziele</b>	Im Studium generale sollen der fachübergreifende Charakter von Lehre und Forschung sowie die Zusammenhänge von Theorie und Praxis vermittelt werden. Der Studierende soll dabei befähigt werden, über sein eigenes Handeln zu reflektieren, sein Wissen einzurichten und Zusammenhänge zu erkennen. Durch die offene und kontroverse Auseinandersetzung anhand eines ausgewählten Themas soll das Urteils- und Handlungsvermögen in politischen, ökonomischen, ökologischen und interkulturellen Bereichen ausgebildet werden.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Keine Angabe
<b>Literaturhinweise</b>	Eine aktuelle Literaturempfehlung erfolgt zu Semesterbeginn durch den/die Dozenten/in.
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Die Form der Lehrveranstaltung kann je nach ausgewähltem Kurs von der Lehrform "Vorlesung" abweichen. Die Anteil der Selbststudienzeit am Workload ist abhängig vom gewählten Kurs.
<b>Verwendbarkeit</b>	in allen Bachelor-Studiengängen
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Mikroprogrammierung und Mikroprozessoren Micropogramming and Microprocessors
<b>Modulnummer</b>	C004 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Praktikum)
<b>Selbststudienzeit</b>	94 Stunden 60 Stunden Bearbeitung Prüfungsleistung 34 Stunden Selbststudium
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Projektarbeit Prüfungsdauer: 3 Monate   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Präsenzvorlesung, Seminar im Labor
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Einfache Hardwarebeschreibungssprachen für kombinatorische und sequenzielle Systeme - Automaten, Mikroprogrammierung und Mikroprogrammsteuerwerke - Mikroprogrammsteuerwerk als didaktisches Modell: Aufbau und Programmierung einer Steuerung, Messung physikalischer Eigenschaften - Mikroprozessoren und Mikrorechner: Zeitverhalten, Adressierungsarten, Befehlsausführung, Interruptsystem, Peripherie Systembauelemente, Leistungsaufnahme und Optimierung
<b>Qualifikationsziele</b>	Die Studentinnen und Studenten sind in der Lage, die verschiedenen Architekturenprinzipien mikroelektronischer Systeme zu charakterisieren und typische Anwendungen mit den hierfür geeigneten Hard- und Softwarewerkzeugen zu implementieren. Die Studenten beherrschen verschiedene Kontrollstrukturen von den Zustandsfolgen endlicher Automaten bis zum Timesharing in Interruptsystemen. Sie können damit Aufgabenstellungen in verteilten und zeitlich parallelen Anwendungen implementieren und Messungen an der Hardware durchführen. Insbesondere sind die Voraussetzungen geschaffen, sich mit Kernel- und Treiberprogrammierung auseinanderzusetzen.
<b>Zulassungsvoraussetzung</b>	keine
<b>Empfohlene Voraussetzungen</b>	Handhabung grundlegender Methoden des Logikentwurfs kombinatorischer Funktionen und endlicher Automaten sowie deren Test in Simulationsumgebungen und in Hardwareanwendungen. Programmiersprache C.

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- Dirk. W. Hoffmann: „Technische Informatik“, Carl Hanser Verlag GmbH &amp; Co. KG; Auflage: 5., 2016, oder später</li> <li>- D. Patterson, J. L. Hennessy: „Rechnerorganisation und Rechnerentwurf: Die Hardware/SoftwareSchnittstelle“, Oldenbourg, 2011, oder später</li> <li>- Wolfram Schiffmann: Technische Informatik 1: Grundlagen der Digitalen Elektronik, Springer, 2003 oder später</li> <li>- Wolfram Schiffmann: Technische Informatik: Übungsbuch zur Technischen Informatik 1 und 2, Springer 2004, oder später</li> </ul>
<b>Aktuelle Lehrressourcen</b>	Skript: Jens Wagner, Theresa Ludwig: Technische Informatik, 2016
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Grundlagen der mobilen Robotik Principles of Mobile Robotics
<b>Modulnummer</b>	C010 Version: 0
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 60 Stunden Bearbeitung Prüfungsleistung 34 Stunden Bearbeitung Prüfungsvorleistung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Laborarbeit
<b>Prüfungsleistung(en)</b>	Prüfung Hausarbeit Prüfungsdauer: 3 Monate   Wichtung: 50%   nicht kompensierbar  Prüfung Referat Prüfungsdauer: 20 Minuten   Wichtung: 50%   nicht kompensierbar
<b>Lehr- und Lernformen</b>	keine Angabe
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Begriffe, Beispielimplementierungen - Einstieg in das wissenschaftliche Arbeiten - Sensorik - Aktoren - Navigation - Roboterkontrollarchitekturen - Designbeispiele: Fußballroboter, Lernroboter, Staubsaugroboter - Projekt und Präsentation
<b>Qualifikationsziele</b>	Die Studierenden sind in der Lage mobile Roboter zu klassifizieren und an Hand ihres Aufbaus zu analysieren. Die Studierenden verfügen über grundsätzliches Wissen zum Aufbau mobiler Roboter und können komplexe praktische Aufgaben im Labor planen und lösen. Einzelne Aspekte werden übergewichtet, um sie im Detail zu behandeln: z.B. Ultraschallsensoren, Kartographie, Wegfindung, Selbstortung. Die Studierenden arbeiten mit aktueller wissenschaftlicher Literatur, einschließlich Monographien und sind in der Lage Ihre eigenen Ergebnisse zu verschriftlichen.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Theoretische und praktische Erfahrungen im Algorithmieren, Programmieren sowie Arbeiten mit Datenstrukturen in einem systemnahen Softwareentwicklungssystem, Beherrschung von physikalischen und logischen Grundlagen der Digitaltechnik, deren Entwurfsmethoden sowie der digitalen Mess- und Analysewerkzeuge. Praktische Erfahrung mit einem einfachen Mikrocontrollerentwicklungssystem.

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- Joachim Hertzberg et al: „Mobile Roboter“, Springer 2012</li> <li>- Ulrich Nehmzow: „Mobile Robotik“, Springer 2013</li> <li>- André Araújo et al. „Integrating Arduino-Based Educational Mobile Robots in ROS“, Journal of Intelligent &amp; Robotic Systems, Volume 77, Issue2, 2015</li> <li>- Saskia Uta Dübener et al.: „Gegenüberstellung von kostengünstigen Robotern als Lernobjekte ...“, Skill – Studierendenkonferenz, Chemnitz, 2017</li> <li>- Anina Ambra Morgner: <a href="https://www.youtube.com/watch?v=ZpcrRbYR64k">https://www.youtube.com/watch?v=ZpcrRbYR64k</a>, 2017</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsprojekt: Individuelle Aufgabenstellungen auf einer einheitlichen, vorgegebenen Plattform. Die Plattform unterliegt jährlichen Verbesserungen, daher ändern sich die Themen regelmäßig.
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Audio-Video-Kommunikation Audio-Video Communication
<b>Modulnummer</b>	C064 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr.-Ing. Jean-Alexander Müller <a href="mailto:jean-alexander.mueller@htwk-leipzig.de">jean-alexander.mueller@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr.-Ing. Jean-Alexander Müller <a href="mailto:jean-alexander.mueller@htwk-leipzig.de">jean-alexander.mueller@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Praktikum   2 SWS Seminar)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung mündliches Fachgespräch Modulprüfung   Prüfungsdauer: 25 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Seminaristische Vorlesungen - Vertiefung von Theorie und anwendungsnahe Vermittlung via E-Learning-System im Praktikum
<b>Medienform</b>	- gestützt auf Präsentationen sowie gemeinsam entwickelte Tafelbilder - Übungsaufgaben zur selbständigen Bearbeitung sowie zur Vorbereitung und Weiterführung der Aufgaben aus dem Praktikum
<b>Lehrinhalte/Gliederung</b>	- Technologische Voraussetzungen - Bedingungen für die multimediale Kommunikation - Kommunikationsmodelle und -dienste - Multimedia – Digitalisierung, Codecs, Präsentation, Systemaufbau - Netzwerk-Technologien für multimediale Kommunikation - Multimediale Kommunikation - Multimediale Anwendungen
<b>Qualifikationsziele</b>	Ziele: detailliertes Fachwissen auf dem Gebiet der multimedialen Kommunikation, zu ihren Einsatzcharakteristika, zu deren Nutzung und zu den Bedingungen / Voraussetzungen eines effektiven Einsatzes detailliertes praxisrelevantes Fachwissen zu einer ausgewählten Spezialrichtung
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Betriebssysteme, Rechnernetze und Programmierung (C)

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- P. L. Dordal: "An Introduction to Computer Networks", ebook, 2018.</li> <li>- R. Steinmetz, K. Nahrstedt: „Multimedia Systems“, Springer 2004.</li> <li>- R. Steinmetz, K. Nahrstedt: „Multimedia Applications“, Springer 2004.</li> <li>- C. Meinel, H. Sack, „Digitale Kommunikation: Vernetzen, Multimedia, Sicherheit“, Springer, 2009.</li> <li>- R. Steinmetz: „Multimedia-Technologie: Grundlagen, Komponenten und Systeme“, Springer, 2000.</li> <li>- W. Effelsberg, R. Steinmetz: „Video Compression Techniques. From JPEG to Wavelets“, dpunkt, 2001.</li> <li>- T. Milde: „Videokompressionsverfahren im Vergleich. JPEG, MPEG, H.261, XCCC, Wavelets, Fraktale“, dpunkt, 1999.</li> <li>- K. Froitzheim: „Multimedia-Kommunikation Dienste, Protokolle und Technik für Telekommunikation und Computernetze“, dpunkt, 1997.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Wahlpflichtmodul: INB (Teil des INB-Bausteins „Technologie für Softwaresysteme“)
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Algorithmische Geometrie Algorithmic Geometry
<b>Modulnummer</b>	C090 Version: 0
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 28 Stunden Selbststudium 56 Stunden Bearbeitung Prüfungsvorleistung 10 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Projektarbeit Prüfungsvorleistung Präsentation
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Vorlesung - Übung - Bearbeitung von Aufgaben und Lösungsfindung - Selbststudium anhand theoretischer und praktischer Übungsaufgaben - studentische Referate
<b>Medienform</b>	Vorlesungsfolien, Übungsblätter, Fachbeiträge
<b>Lehrinhalte/Gliederung</b>	- Berechnung von Segmentschnitten - Triangulierung von Polygonen - Effiziente Suchstrukturen - Punktsuche - Voronoi-Diagramme - Delaunay-Triangulierung - Berechnung der konvexen Hülle - Pfadsuche und Bewegungsplanung  ausgewählte Beispiele aus verschiedenen Anwendungsgebieten, z.B. in autonomen Fahrzeugen
<b>Qualifikationsziele</b>	Die Studierenden können praktische Probleme auf geometrische Fragestellungen zurückführen. Sie können geometrische Probleme beurteilen und geeignete Datenstrukturen und Algorithmen zu deren Lösung einsetzen.  Sie sind in der Lage, sich anhand aktueller Fachliteratur eigenständig in Anwendungen auf verschiedenen Gebieten einzuarbeiten, ihre Erkenntnisse zu strukturieren und anzuwenden.
<b>Zulassungsvoraussetzung</b>	Keine

<b>Empfohlene Voraussetzungen</b>	anwendungsbereite Kenntnisse auf den - Gebieten Modellierung, - Algorithmen und Datenstrukturen, - lineare Algebra, - Aufwandsabschätzungen
<b>Literaturhinweise</b>	de Berg, M. et al: Computational Geometry, Algorithms and Applications. Springer 2008 Klein, R.: Algorithmische Geometrie. Springer 2005 Joswig, M. und T. Theobald: Algorithmische Geometrie. Vieweg 2007 Preparata, F. P. und M. I. Shamos: Computational Geometry. Springer 1993
<b>Aktuelle Lehrressourcen</b>	Lehrmaterial und aktuelle Informationen: <a href="https://informatik.htwk-leipzig.de/schwarz">https://informatik.htwk-leipzig.de/schwarz</a>
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor Wahlpflichtmodul Medieninformatik   Bachelor Wahlpflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	<a href="https://informatik.htwk-leipzig.de/schwarz">https://informatik.htwk-leipzig.de/schwarz</a>

<b>Modul</b>	Computergrafik Computer Graphics
<b>Modulnummer</b>	C121 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>  Prof. Dr. Kiran Varanasi <a href="mailto:kiran.varanasi@htwk-leipzig.de">kiran.varanasi@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 60 Stunden Bearbeitung Prüfungsvorleistung 34 Stunden Selbststudium
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung am Computer
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Einsatz von e-Learning mit automatisierten Tests zur Eigenkontrolle
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Gerätetechnik - Algorithmen der Computergrafik - Geometrische Transformationen - Visualisierung - Datenmodelle für geometrische Objekte
<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss der Lehrveranstaltung sind die Studierenden in der Lage Methoden der generativen Computergrafik wie Modellierung, Transformation und Visualisierung von geometrischen Objekten in Projekten einzusetzen.  Sie können die Stärken und Schwächen der geometrischen Modelle sowie ihre Einsatzmöglichkeiten einschätzen und beherrschen die entsprechenden mathematischen Grundlagen.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Programmieren in einer Objektorientierten Programmiersprache, Analytische Geometrie, Lineare Algebra
<b>Literaturhinweise</b>	- Ein Skript oder Folien der Vorlesungen werden in OPAL zur Verfügung gestellt. - Ergänzende aktuelle Literatur zur Vorlesung findet sich in OPAL.
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistung am Computer (PVC): Bearbeitung einer Praktikumsaufgabe und Präsentation der Ergebnisse am Computer.

<b>Verwendbarkeit</b>	Medieninformatik   Bachelor (20MIB) Pflichtmodul Informatik   Bachelor (20INB) Wahlpflicht Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Constraint-Programmierung Constraint Programming
<b>Modulnummer</b>	C160 Version: 3
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Nach Bekanntgabe der Fakultät
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Vorlesung - Übung - E-Learning (automatische Bewertung eines Teiles der Hausaufgaben)
<b>Medienform</b>	- Tafelanschrieb - Skript
<b>Lehrinhalte/Gliederung</b>	- Aussagenlogische Constraints • Syntax, Semantik, Normalformen, Tseitin-Transformation • DPLL-Solver, Conflict Driven Clause Learning • Binäre Entscheidungsdiagramme  Prädikatenlogische Constraints • Termgleichungen, Unifikation, • lineare Gleichungen und Ungleichungen über reellen und ganzen Zahlen • Polynomgleichungen, Presburger-Arithmetik  Kombinationen • Nelson-Oppen-Verfahren für konvexe Theorien • SMT mit DPLL(T) • SMT mit SAT-Kodierungen (Bit Blasting)
<b>Qualifikationsziele</b>	Studenten kennen Modelle, Methoden und Werkzeuge der Constraint-Programmierung, können Anwendungsaufgaben als Constraint-Probleme formulieren und durch geeignete Verfahren lösen, können Aufwand der Lösungsalgorithmen richtig einschätzen.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Kenntnisse der Algebra und der Prädikatenlogik
<b>Literaturhinweise</b>	K. Apt: „Principles of Constraint Programming“, Cambridge Univ. Press, 2003. D. Kroening, O. Strichman: „Decision Procedures“, Springer, 2008. P. Hofstedt, A. Wolf: „Einführung in die Constraint-Programmierung“, Springer, 2007.

<b>Aktuelle Lehrressourcen</b>	- keine
<b>Hinweise</b>	Prüfungsvorleistung Beleg (PVB): Regelmäßiges und erfolgreiches Bearbeiten von Übungsaufgaben
<b>Verwendbarkeit</b>	Informatik   Master (20INM) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Computeranimation Computer Animation
<b>Modulnummer</b>	C182 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung mündliches Fachgespräch Prüfungsdauer: 20 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Programmierbeispiele und Erarbeiten von Animationslösungen in den Seminaren
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Grundlagen der Computeranimation</li> <li>- Herstellung einer Computeranimation</li> <li>- Animationstechniken</li> <li>- Rendering</li> <li>- Erstellung von Spezialeffekten</li> </ul>
<b>Qualifikationsziele</b>	<p>Nach erfolgreichem Abschluss beherrschen die Studierenden Grundtechniken der 3D-Modellierung von Szenen mit Körpern als polygonale Netze, Prinzipien verschiedener Beleuchtungsverfahren und den Einsatz von Kameras. Sie beherrschen Verfahren der Computeranimation wie KeyframeAnimation, Methoden der inversen Kinematik, Motion Capture und Morphing. Durch Einsatz von Materialien und Mapping-Techniken sind sie in der Lage, die erstellten Szenen mit verschiedenen Renderverfahren fotorealistisch präsentieren.</p> <p>Die Studierenden setzen diese Kenntnisse in einem kommerziellen Computeranimationssystem bis zur Fertigstellung einer Computeranimation exemplarisch um. Sie sind in der Lage den Einsatz der Software für verschiedene Anwendungen einzuschätzen.</p>
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Grundlagen der Darstellenden Geometrie, Vorlesung Computergrafik (empfohlen), Programmierkenntnisse
<b>Literaturhinweise</b>	<p>Ein Skript oder Folien der Vorlesungen werden im OPAL zur Verfügung gestellt</p> <p>Ergänzende aktuelle Literatur zur Vorlesung findet sich im OPAL</p> <p>Die Vorlesung bezieht sich in Auszügen auf:</p> <p>A. H. Watt, M. Watt, "Advanced animation and rendering techniques: Theory and practice(Reprint.)", New York, NY, ACM Press, 1998.</p>

<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Wahlpflicht Medieninformatik   Bachelor (20MIB) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Vertiefende Spezialvorlesung der Informatik 1 Special Topics in Computer Science
<b>Modulnummer</b>	C227 Version: 1
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr.-Ing. Robert Müller <a href="mailto:robert.mueller@htwk-leipzig.de">robert.mueller@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. habil. Michael Frank <a href="mailto:michael.frank@htwk-leipzig.de">michael.frank@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>  Prof. Dr.-Ing. Thomas Kudraß <a href="mailto:thomas.kudrass@htwk-leipzig.de">thomas.kudrass@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Thomas Riechert <a href="mailto:thomas.riechert@htwk-leipzig.de">thomas.riechert@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>  Prof. Dr. Kiran Varanasi <a href="mailto:kiran.varanasi@htwk-leipzig.de">kiran.varanasi@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Karsten Weicker <a href="mailto:karsten.weicker@htwk-leipzig.de">karsten.weicker@htwk-leipzig.de</a>  Prof. Dr.-Ing. Jean-Alexander Müller <a href="mailto:jean-alexander.mueller@htwk-leipzig.de">jean-alexander.mueller@htwk-leipzig.de</a>  Dr. rer. nat. Sebastian Rinke <a href="mailto:sebastian.rinke@htwk-leipzig.de">sebastian.rinke@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Testat
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Vorlesung und für das jeweilige Thema angepasste Seminarübungen - je nach Dozentenwunsch individuell oder in Gruppen.
<b>Medienform</b>	keine Angabe

<b>Lehrinhalte/Gliederung</b>	Die Lehrinhalte wechseln abhängig vom jeweiligen Dozenten. Grundsätzlich wird ein Gebiet der Informatik über die bestehenden Lehrveranstaltungsangebote hinaus vertieft.
<b>Qualifikationsziele</b>	Die Studierenden erwerben Kenntnisse in einem vertiefenden Thema der Informatik, welches nicht bereits durch andere Veranstaltungen im Bachelorstudiengang abgedeckt wird. Sie können diese Kenntnisse auf kleine Beispiele wie auch in einem größeren Anwendungskontext anwenden.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Die Voraussetzungen werden für die einzelnen Lehrveranstaltungsangebote separat bei der Ankündigung veröffentlicht.
<b>Literaturhinweise</b>	Literatur wird themenabhängig vom Dozenten bekannt gegeben.
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistung: regelmäßige Mitarbeit in den Seminaren
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Wahlpflicht  Medieninformatik   Bachelor (20MIB) Wahlpflicht  Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Grundlagen Internet-basierter Informationssysteme Introduction to Internet Based Information Systems
<b>Modulnummer</b>	C247 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Thomas Riechert <a href="mailto:thomas.riechert@htwk-leipzig.de">thomas.riechert@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Thomas Riechert <a href="mailto:thomas.riechert@htwk-leipzig.de">thomas.riechert@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung mündliches Fachgespräch Prüfungsdauer: 30 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	keine Angabe
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Einführung – Geschichte und Struktur des Internets</li> <li>- Einführung – Verteilte Informationssysteme</li> <li>- Internet-Stack, Infrastruktur</li> <li>- Applikationsschicht (Ausgewählte Anwendungen)</li> <li>- HTTP-Protokoll</li> <li>- Web-Architekturen</li> <li>- Service Orientierte Architekturen (SOA), Webservices</li> <li>- JSON-REST-Services</li> <li>- Semantic Web</li> <li>- Verteilte Informationsverarbeitung</li> </ul> <p>Im Rahmen der Übungen werden die Inhalte der Vorlesung in praktischen Experimenten nachvollzogen. Dabei werden u.a. ein Unix-Server installiert, verschiedene Webapplikationen installiert und getestet, sowie Schnittstellen definiert und entwickelt.</p>
<b>Qualifikationsziele</b>	Nach erfolgreichem Besuch der Veranstaltung sind die Studierenden in der Lage Protokolle und Systemkomponenten für die Kommunikation über Internetverbindungen zu beurteilen und auszuwählen. Sie können damit auf der Basis von TCP und UDP verteilte Anwendungen und Schnittstellen für Internet-basierte Informationssysteme entwickeln.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Die Studierenden beherrschen den Aufbau und die Arbeitsweise von Rechnernetzen und die darin eingesetzten Protokollhierarchien.
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- Ch. Meinel, H. Sack: „Internetworking: Technische Grundlagen und Anwendungen“, Springer, 2012.</li> <li>- A. S. Tanenbaum, D. Wetherall: „Computernetzwerke“, Pearson, 2012.</li> <li>- Weiterführende Literaturhinweise werden zu Beginn der Lehrveranstaltung bekanntgegeben.</li> </ul>

<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor Wahlpflichtmodul Medieninformatik   Bachelor Wahlpflichtmodul Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik Wahlpflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Semantic Web Semantic Web
<b>Modulnummer</b>	C249 Version: 3
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Thomas Riechert <a href="mailto:thomas.riechert@htwk-leipzig.de">thomas.riechert@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Thomas Riechert <a href="mailto:thomas.riechert@htwk-leipzig.de">thomas.riechert@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Seminar)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Projektarbeit Prüfungsdauer: 6 Wochen   Wichtung: 100%
<b>Lehr- und Lernformen</b>	-
<b>Medienform</b>	Keine Angabe
<b>Lehrinhalte/Gliederung</b>	- - RDF-Datenmodell - Web Ontology Language (OWL) - Regeln im Semantic Web - SPARQL als Abfragesprache für RDF - Linked Open Data - Linked Enterprise Data - Web- und datenbasiertes Knowledge Engineering (Ontology Learning, Datenqualität)
<b>Qualifikationsziele</b>	Die Studierenden erlangen fundierte Kenntnisse zu den Grundlagen, Technologien und Anwendungen des Semantic Web. Sie erwerben die Fähigkeit Semantic Web Technologien und Werkzeuge praktisch anzuwenden und deren Einsatzmöglichkeiten für gegebene Problemstellungen einzuschätzen. An semantisch orientierten Methoden beherrschen die Studierenden insbesondere die die semantische Suche im Web, Grundlagen des Web- und datenbasierten Knowledge Engineering und Methoden der Informationsintegration.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Gutes Verständnis relationaler SQL-Datenbanken, eine objektorientierte Programmiersprache (z.B. Java oder C#), Grundverständnis für Webarchitekturen und deren Schnittstellen (HTTP-Protokoll, HTML, XML)

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- T. Berners-Lee, J. Hendler, Ora Lassila: „The Semantic Web: a new form of Web content that is meaningful to computers will unleash a revolution of new possibilities“, In: <i>Scientific American</i>, 284(5), S. 34–43, 2001 (dt.: Mein Computer versteht mich. In: <i>Spektrum der Wissenschaft</i>, August 2001, S. 42–49), <a href="http://www.cs.umd.edu/~golbeck/LBSC690/SemanticWeb.html">http://www.cs.umd.edu/~golbeck/LBSC690/SemanticWeb.html</a></li> <li>- P. Hitzler, M. Krötzsch, S. Rudolph, Y. Sure: „Semantic Web. Grundlagen.“, Springer Verlag, 2008, <a href="http://www.semantic-web-book.org">http://www.semantic-web-book.org</a></li> <li>- Resource Description Framework (RDF): <a href="http://www.w3.org/RDF/">http://www.w3.org/RDF/</a></li> <li>- W3C Recommendation RDF-Schema 1.0: <a href="http://www.w3.org/TR/rdf-schema/">http://www.w3.org/TR/rdf-schema/</a></li> <li>- Web Ontology Language (OWL): <a href="http://www.w3.org/OWL/">http://www.w3.org/OWL/</a></li> </ul>
<b>Aktuelle Lehrressourcen</b>	<ul style="list-style-type: none"> <li>-</li> <li>keine</li> </ul>
<b>Hinweise</b>	<ul style="list-style-type: none"> <li>-</li> </ul>
<b>Verwendbarkeit</b>	<ul style="list-style-type: none"> <li>Informatik   Master Wahlpflichtmodul</li> <li>Medieninformatik   Master Wahlpflichtmodul</li> </ul>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Thread-Programmierung Thread Programming
<b>Modulnummer</b>	C259 [C529] Version: 3
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. habil. Alfons Geser <a href="mailto:alfons.geser@htwk-leipzig.de">alfons.geser@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. habil. Alfons Geser <a href="mailto:alfons.geser@htwk-leipzig.de">alfons.geser@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Testat
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	-
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- - Prozesse, Threads, Parallelität, Verschränkung - Gemeinsamer Speicher, Zugriffskonflikte, Synchronisation, Verklemmung - Thread-Sicherheit, map/reduce - Verifikation von Systemen mit Threads - Programmierung auf Graphikprozessoren
<b>Qualifikationsziele</b>	Die Teilnehmer kennen Ausdrucksmittel für parallele und nebenläufige Programme in verschiedenen Programmierparadigmen und –sprachen und können mit Threads und den gängigen Synchronisationstechniken umgehen. Sie kennen die Probleme, die bei der Programmierung mit gemeinsamen Ressourcen auftreten können, und können Methoden zu ihrer Vermeidung anwenden. Sie sollen einen Einblick bekommen in Verifikation und GPU-Programmierung.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Kenntnisse in Paralleler Programmierung und Betriebssystemen
<b>Literaturhinweise</b>	C. Lin, L. Snyder: „Principles of Parallel Programming“, Addison Wesley, 2009.
<b>Aktuelle Lehrressourcen</b>	- keine
<b>Hinweise</b>	Prüfungsvorleistung Testat (PVT): Bearbeitung und Präsentation von Übungsaufgaben im Seminar
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Prozessautomatisierung Automation of Technical Processes
<b>Modulnummer</b>	C383 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Praktikum)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung mündliches Fachgespräch Prüfungsdauer: 30 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	keine Angabe
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Automatisierung technischer Prozesse</li> <li>- Stetige und binäre Steuerungen</li> <li>- Speicherprogrammierbare Steuerungen</li> <li>- Regelungen und Fuzzy Control</li> <li>- Neuronale Konzepte und Neuro-Fuzzy-Control</li> </ul>
<b>Qualifikationsziele</b>	Die Studierenden können nach dem erfolgreichen Abschluss des Moduls, bestimmte technische Prozesse durch den Einsatz von informationsverarbeitenden Systemen automatisieren. Dazu werden insbesondere für verschiedene Aufgabenklassen Steuerungen und Regelungen entworfen und diese in entsprechende Programme umgesetzt und getestet. Dabei kommen insbesondere SPSEN zum Einsatz, auf deren Grundlage verschiedene Programmierungsmöglichkeiten genutzt werden.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Kenntnisse und Fertigkeiten zum Entwurf von Schaltnetzen und Schaltwerken
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- R. Langmann: „Taschenbuch der Automatisierung“, Fachbuchverlag Leipzig, aktuelle Auflage.</li> <li>- R. Lauber, P. Göhner: „Prozessautomatisierung“, Springer, aktuelle Auflage.</li> <li>- M. Seitz: „Speicherprogrammierbare Steuerungen“, Fachbuchverlag Leipzig, aktuelle Auflage.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor Wahlpflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Hardware-Entwurfstechnik Hardware Design Tools
<b>Modulnummer</b>	C448 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 34 Stunden Vorbereitung Lehrveranstaltung 60 Stunden Bearbeitung Prüfungsvorleistung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Projektarbeit
<b>Prüfungsleistung(en)</b>	Prüfung mündliches Fachgespräch Prüfungsdauer: 30 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	keine Angabe
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Manuell entworfene Komponenten</li> <li>• Addierer</li> <li>• Multiplizierer</li> <li>• Dividierer</li> <li>- Logiksynthese</li> <li>• Zweistufige Logikminimierung</li> <li>• Mehrstufige Logiksynthese</li> <li>- Entwurf von Steuerwerken</li> <li>• Architekturen von Automaten</li> <li>• Zustandscodierung</li> <li>- Einführung in VHDL</li> </ul>
<b>Qualifikationsziele</b>	Die Studenten können verschiedene Entwurfsansätze auf der RT-, Logikebene sowie die Arbeitsweise der Entwurfssysteme nachvollziehen. Sie können zu einem gegebenen Problem eine Hardware-Lösung spezifizieren und (insbesondere mit FPGAs) realisieren.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Entwurf digitaler Schaltungen
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- M. Ercegovac, T. Lang: „Digital Arithmetic“, Morgan Kaufmann Publishers, 2003.</li> <li>- M. Lu: „Arithmetic and Logic in Computersystems“, Wiley, 2004.</li> <li>- J. Reichardt, B. Schwarz: „VHDL-Synthese: Entwurf digitaler Schaltungen und Systeme“, Oldenbourg, 2012.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	Foliensatz
<b>Hinweise</b>	Keine Angabe

<b>Verwendbarkeit</b>	Informatik   Bachelor Wahlpflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Einführung in die virtuelle und erweiterte Realität (VR/AR) Introduction to Virtual and Augmented Reality
<b>Modulnummer</b>	C493 [C384] Version: 0
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. Kiran Varanasi <a href="mailto:kiran.varanasi@htwk-leipzig.de">kiran.varanasi@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. Kiran Varanasi <a href="mailto:kiran.varanasi@htwk-leipzig.de">kiran.varanasi@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 64 Stunden Selbststudium 30 Stunden Bearbeitung Prüfungsleistung
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Projektarbeit Prüfungsdauer: 6 Wochen   Wichtung: 50%   nicht kompensierbar  Prüfung Referat Prüfungsdauer: 20 Minuten   Wichtung: 50%   nicht kompensierbar
<b>Lehr- und Lernformen</b>	keine Angabe
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ol style="list-style-type: none"> <li>1. Grundlagen und Begriffsbestimmung zu VR und AR</li> <li>2. Ausgabeperipherie <ul style="list-style-type: none"> <li>- Lichtfelder und plenoptisches Rendering</li> <li>- Stereoskopisches Sehen und technische Umsetzung</li> <li>- Dreidimensionales Hören und technische Umsetzung</li> <li>- Haptische Rendering</li> </ul> </li> <li>3. Eingabeperipherie <ul style="list-style-type: none"> <li>- Kamera-Tracking und visuelle Odometrie</li> <li>- Motion Capture: Kopf, Hände, Körper</li> <li>- Eye tracking</li> <li>- Wearables: Activity Trackers</li> </ul> </li> <li>4. Räumliche Kognition und Navigation in 3D</li> <li>5. 3D Modellierung und Interaktionsdesign</li> <li>6. Beispiele für VR-Systeme</li> <li>7. Beispiele für AR-Systeme</li> </ol> <p>Praktische Übungen zur Gestaltung und Realisierung interaktiver virtueller Welten und zur interaktiven Steuerung von Objekten.</p>
<b>Qualifikationsziele</b>	Die Studierenden beherrschen die Entwicklung und Gestaltung von virtuellen Welten unterschiedlichen Immersionsgrades. Sie besitzen Grundkenntnisse zum Aufbau der Hardwarekomponenten verschiedener VR-Systeme und AR-Systeme. Entwurf und Programmierung interaktiver virtueller Welten werden eingeübt.

<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Mathematische und physikalische Grundkenntnisse auf Abiturniveau
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- F. Eckgold: "Virtual Reality", Vieweg &amp; Sohn, 1995.</li> <li>- M. Brill: "Virtuelle Realität (Informatik im Fokus)", Springer, 2008</li> <li>- R. Dörner, W. Broll, P. Grimm, B. Jung "Virtual und Augmented Reality: Grundlagen und Methoden der Virtuellen und Augmentierten Realität", Springer, 2019</li> <li>- D. Korgel: "Virtual-Reality-Spieleentwickeln mit Unity", Hanser Verlag 2018</li> <li>- D. Schmalstieg &amp; T. Höllerer „Augmented reality: Principles and Practice“ Addison-Wesley, 2016</li> <li>- J. Linowes, K. Babilinski „Augmented Reality for developers: Build practical augmented reality applications with Unity, ARCore, ARKit, and Vuforia“ Packt, 2017</li> <li>- D. Scherfgen: "3D Spieleprogrammierung mit DirectX 9 und C++", Carl Hanser Verlag, 2006.</li> <li>- S. Wigard: "Spieleprogrammierung mit DirectX 11 und C++", Hüthig, 2010.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	<p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Informatik   Bachelor (20INB) Wahlpflicht</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	e-Learning e-Learning
<b>Modulnummer</b>	C585 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Thomas Riechert <a href="mailto:thomas.riechert@htwk-leipzig.de">thomas.riechert@htwk-leipzig.de</a>
<b>Dozierende</b>	Felix Stolze
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 30 Stunden Vorbereitung Lehrveranstaltung 64 Stunden Bearbeitung Prüfungsvorleistung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Projektarbeit
<b>Prüfungsleistung(en)</b>	Prüfung mündliches Fachgespräch Modulprüfung   Prüfungsdauer: 30 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	- Präsentation (Vorlesung) - Individuelle technische Übung - Kollektive Projektarbeit
<b>Medienform</b>	- Folienpräsentation - Elektronisch bereitgestellte Übungsblätter - Literatur - Internet-Quellen
<b>Lehrinhalte/Gliederung</b>	<p>1. Begriffsbestimmung Lernen und Lehren, Geschichte, Lerntheorien, E-Learning, Szenarien, Lernmanagement, Content</p> <p>2. Potenzial, Probleme und Entwicklung Aktuelle Entwicklungslinien, Programme und Initiativen, Projekte</p> <p>3. Konzeption von E-Learning-Angeboten Instruktionsdesign, Strukturierung des Vorgehens, Didaktik</p> <p>4. Analyse und Planung Zielgruppenanalyse, Wahl der Lehr-/Lernmethode, adäquater Medieneinsatz</p> <p>5. Entwicklung und Produktion Werkzeugeinsatz, Rapid E-Learning, Text- und Bildgestaltung</p> <p>6. Ausgewählte Aspekte Evaluation, Standardisierung, Open Educational Resources, ...</p> <p>In den Übungen werden aktuelle Werkzeuge zur Erstellung von E-Learning-Szenarien getestet und das als Prüfungsvorleistung geforderte Projekt vorbereitet.</p>

<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss der Lehrveranstaltung haben die Studierenden ein fundiertes Verständnis von E-Learning als einem interdisziplinären Fachgebiet im Schnittpunkt von Informatik, Didaktik und multimedialem Design. Sie begreifen E-Learning-Szenarien als sinnvolle Ergänzung traditioneller Lehr- und Lernformen und können Probleme und Potential des E-Learning bezogen auf den Hochschulbereich diskutieren. Sie sind in der Lage, aktuelle Entwicklungen auf dem Gebiet einzuschätzen. Die Studierenden sind mit ausgewählten Werkzeugen zur Realisierung von E-Learning-Szenarien vertraut. Sie verfügen über die technischen und didaktischen Fähigkeiten, Lernmodule zielgruppengerecht zu konzipieren und umzusetzen. Des Weiteren sind sie in der Lage, adäquate Evaluationsmethoden zum Einsatz zu bringen.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Grundfertigkeiten bei der Erzeugung digitaler Medienobjekte im visuellen und auditiven Bereich, Erfahrungen der vielfältigen Umsetzbarkeit von Lehrveranstaltungen im Hochschulbereich aus Lernersicht
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- M. Ebner, S. Schön: "L3T: Lehrbuch für Lernen und Lehren mit Technologien", <a href="http://l3t.eu/homepage/das-buch/ebook-2013">http://l3t.eu/homepage/das-buch/ebook-2013</a>.</li> <li>- H. M. Niegemann et al.: "Kompendium multimediales Lernen", Springer, 2008.</li> <li>- M. Kerres: "Mediendidaktik: Konzeption und Entwicklung mediengestützter Lernangebote", 5. Aufl., Walter de Gruyter, 2018.</li> <li>- G. Siemens: "Connectivism: A Learning Theory for the Digital Age", <i>International Journal of Instructional Technology &amp; Distance Learning</i>, Vol. 2 No.1, 2005.</li> <li>- H. Fischer, J. Schwendel: "E-Learning an sächsischen Hochschulen; Strukturen - Projekte - Einsatzszenarien", TUDpress, 2009.</li> <li>- <a href="https://www.e-teaching.org/">https://www.e-teaching.org/</a></li> <li>- <a href="https://www.toptools4learning.com/">https://www.toptools4learning.com/</a></li> </ul> <p>Diverse Schrift- und Internet-Quellen je nach Thematik und Zeitraum.</p>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	<p>Vorlesungsfolien, Übungsmaterial, Beispiele, aktuelle Quellen und Informationen zur Veranstaltung werden im Laufe des Semesters über das LMS OPAL bereitgestellt.</p> <p>Die Lösungsabgabe zu den Übungsaufgaben erfolgt ebenfalls über OPAL.</p>
<b>Verwendbarkeit</b>	<p>Medieninformatik   Bachelor Wahlpflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik Wahlpflichtmodul</p> <p>Informatik   Bachelor Wahlpflichtmodul</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Dokumentbeschreibungssprachen Document Description Languages
<b>Modulnummer</b>	C651 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. habil. Michael Frank <a href="mailto:michael.frank@htwk-leipzig.de">michael.frank@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. habil. Michael Frank <a href="mailto:michael.frank@htwk-leipzig.de">michael.frank@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Seminar)
<b>Selbststudienzeit</b>	94 Stunden 34 Stunden Selbststudium 60 Stunden Bearbeitung Prüfungsleistung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Projektarbeit Modulprüfung   Prüfungsdauer: 6 Wochen   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Einzel- und Gruppenarbeit am Projekt
<b>Medienform</b>	-
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Einführung in XML als Markup-, Datentransport- und als Applikationssteuerungssprache, Einführung in XML-Editoren und Parser</li> <li>- Wohlgeformtheit und Gültigkeit von Dokumenten</li> <li>- Strukturdefinition mit Document Type Definition (DTD)</li> <li>- Darstellung von XML-Inhalten als Webseiten mit CSS</li> <li>- Darstellung von XML-Inhalten als textbasierte Dateien mit XSLT, Dokumentformate</li> <li>- Strukturdefinitionen mit XML-Schema-Definitionen und ihre verschiedenen Designs</li> <li>- Einführungen in XSL-FO und Schematron</li> <li>- Praktische Übungen aller Aspekte, großes Projekt zum Datentransport und zur Datendarstellung, Beispielapplikationen aus der Praxis</li> </ul>
<b>Qualifikationsziele</b>	Syntax und Semantik der eXtensible Markup Language (XML), ihrer Strukturdefinitionen Document Type Definition (DTD) und XML-Schema Definition (XSD) und der Darstellungssprache eXtensible Stylesheet Language (XSLT-Fall) werden beherrscht. Anhand eines umfangreichen Programmierprojekts wurden praktische Erfahrungen mit XML-Projekten erworben. Im Umgang mit LaTeX als einer möglichen Umsetzungsform großer Dokumente sind für die Bachelorarbeit anwendbare Fertigkeiten entstanden.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Beherrschung statischer Webprogrammierung mit HTML und CSS
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- H. Erlenkötter: "XML - Extensible Markup Language von Anfang an", Rowohlt, 2003.</li> <li>- H. Vonhoegen: „Einstieg in XML: Grundlagen, Praxis, Referenz. Das XML-Handbuch mit vielen Anwendungsbeispielen“, Galileo Computing, 8. Auflage, 2015/2018.</li> <li>- W. Grupe: „XML: Grundlagen   Technologien  Validierung   Auswertung“, mitp, 2018.</li> <li>- F. Bongers: „XSLT 2.0 und XPath 2.0“, Galileo Computing, 2008.</li> <li>- M. Krüger, U. Welsch: „XSL-FO Praxis. Schnell &amp; kompakt“, Entwickler.Press, 2007/2015.</li> <li>- Spezifikationen des W3C zu den XML-Standards, weitere Empfehlungen im Kurs.</li> </ul>

<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistung Belege (PVB): Übungsfragen und -aufgaben (wöchentlich)
<b>Verwendbarkeit</b>	Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul Informatik   Bachelor (20INB) Wahlpflicht Medieninformatik   Bachelor (20MIB) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Mobile Computing Mobile Computing
<b>Modulnummer</b>	C652 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Nach Bekanntgabe der Fakultät
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Andreas Both
<b>Dozierende</b>	Prof. Dr. rer. nat. Uwe Petermann <a href="mailto:uwe.petermann@htwk-leipzig.de">uwe.petermann@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung am Computer Modulprüfung   Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Vorlesung  projektorientierte Seminare im Computerpool
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Kommunikationsprotokolle für mobile Anwendungen.</li> <li>- Programmier-Plattformen für mobile Anwendungen (insbesondere Java Micro Edition, Android, IOS, weitere).</li> <li>- Techniken und Werkzeuge der Cross-Plattform-Entwicklung.</li> <li>- Sicherheitsaspekte bei Endgeräten, Kommunikation und Anwendungen</li> <li>- Praktische Übungen zur Konzeption und Realisierung von Anwendungen des Mobile Computing.</li> </ul>
<b>Qualifikationsziele</b>	Die Studierenden sind zur Konzeption und zur Entwicklung von Anwendungslösungen mit mobilen Kommunikationsgeräten der wichtigsten Plattformen befähigt. Sie beherrschen die aktuellen Standards und Kommunikationsprotokolle sowie die Programmierplattformen für mobile Endgeräte.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Anwendungsbereite Kenntnisse zu Hard- und Software von Rechnern und Netzen; Beherrschung der Entwicklung von Lösungen für Praxisprobleme unter Verwendung höherer Programmiersprachen;  Befähigung zur Auswahl und zum Einsatz der für die Lösung von Praxisproblemen geeigneten Algorithmen und Datenstrukturen, sowie Werkzeuge.
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- M. Ross: "PhoneGap - Mobile Cross-Plattform-Entwicklung", dpunkt-Verlag, 2013.</li> <li>- J. Stark: "Building Android Apps with HTML, CSS, and JavaScript", O'Reilly, 2012.</li> <li>- U. Post: "Android-Apps entwickeln", Galileo Computing, 2012.</li> <li>- J. Roth: "Mobile Computing", dpunkt-Verlag, 2005.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe

<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Wahlpflicht</p> <p>Medieninformatik   Bachelor (20MIB) Wahlpflicht</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Wahlpflicht</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Vertiefende Spezialvorlesung der Informatik 2 Special Topics in Computer Science
<b>Modulnummer</b>	C672 Version: 1
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Studiendekan (INB, INM)
<b>Dozierende</b>	Prof. Dr. rer. nat. habil. Michael Frank <a href="mailto:michael.frank@htwk-leipzig.de">michael.frank@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>  Prof. Dr.-Ing. Thomas Kudraß <a href="mailto:thomas.kudrass@htwk-leipzig.de">thomas.kudrass@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Thomas Riechert <a href="mailto:thomas.riechert@htwk-leipzig.de">thomas.riechert@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Sibylle Schwarz <a href="mailto:sibylle.schwarz@htwk-leipzig.de">sibylle.schwarz@htwk-leipzig.de</a>  Prof. Dr. Kiran Varanasi <a href="mailto:kiran.varanasi@htwk-leipzig.de">kiran.varanasi@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>  Prof. Dr. rer. nat. Karsten Weicker <a href="mailto:karsten.weicker@htwk-leipzig.de">karsten.weicker@htwk-leipzig.de</a>  Prof. Dr.-Ing. Jean-Alexander Müller <a href="mailto:jean-alexander.mueller@htwk-leipzig.de">jean-alexander.mueller@htwk-leipzig.de</a>  Dr. rer. nat. Sebastian Rinke <a href="mailto:sebastian.rinke@htwk-leipzig.de">sebastian.rinke@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Testat
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Vorlesung und für das jeweilige Thema angepasste Seminarübungen - je nach Dozentenwunsch individuell oder in Gruppen.
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	Die Lehrinhalte wechseln abhängig vom jeweiligen Dozenten. Grundsätzlich wird ein Gebiet der Informatik über die bestehenden Lehrveranstaltungsangebote hinaus vertieft.

<b>Qualifikationsziele</b>	Die Studierenden erwerben Kenntnisse in einem vertiefenden Thema der Informatik, welches nicht bereits durch andere Veranstaltungen im Bachelorstudiengang abgedeckt wird. Sie können diese Kenntnisse auf kleine Beispiele wie auch in einem größeren Anwendungskontext anwenden.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Die Voraussetzungen werden für die einzelnen Lehrveranstaltungsangebote separat bei der Ankündigung veröffentlicht.
<b>Literaturhinweise</b>	Literatur wird themenabhängig vom Dozenten bekannt gegeben.
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistung: regelmäßige Mitarbeit in den Seminaren
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Wahlpflicht Medieninformatik   Bachelor (20MIB) Wahlpflicht Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Datenbanken (Aufbaukurs) Database Systems (Advanced Level)
<b>Modulnummer</b>	C720 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr.-Ing. Thomas Kudraß <a href="mailto:thomas.kudrass@htwk-leipzig.de">thomas.kudrass@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr.-Ing. Thomas Kudraß <a href="mailto:thomas.kudrass@htwk-leipzig.de">thomas.kudrass@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Testat
<b>Prüfungsleistung(en)</b>	Prüfung mündliches Fachgespräch Prüfungsdauer: 30 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Die Teilnehmer lösen auf der Grundlage der in der Vorlesung vermittelten Inhalte im Rahmen von Rechnerübungen verschiedene Programmieraufgaben. Dabei geht es darum, die Datenbankprogrammierung in kleinen Anwendungen zu trainieren bzw. alternative und erweiterte Datenbankkonzepte, die über das Relationenmodell hinausreichen, an praktischen Beispielen zu vertiefen. Dabei kommen unterschiedliche Datenbankschnittschnellen und Datenbankmodelle zum Einsatz, so dass die Teilnehmer deren Charakteristik aus eigener Anschauung besser beurteilen können. Die Übungsaufgaben sind selbstständig zu absolvieren, werden aber vom Seminarleiter bewertet. Somit entspricht die regelmäßig testierte Teilnahme an den Übungen der Prüfungsvorleistung.
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Datenbank-Anwendungsprogrammierung mit PL/SQL (Oracle) - Objektrelationale und objektorientierte Datenbanken - Java und Datenbanken (JDBC, Java Persistence API) - XML und Datenbanken: Speicherung von XML, Anfragesprachen (XML/SQL, XQuery) - NoSQL-Datenbanken - Datenbanken im Web (Anwendungen, Systemarchitekturen, DB-Zugriffsschnittstellen)
<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss des Moduls hat der Student umfangreiche Erfahrungen bei der Entwicklung von Datenbankprojekten. Er kann die Konzepte einer Datenbankprogrammiersprache bei der Lösung von praktischen Programmieraufgaben anwenden. Der Student kennt eine Reihe von Datenbankmodellen, die das Relationenmodell erweitern bzw. alternativ dazu gesehen werden können und kann deren Merkmale für bestimmte Anwendungen bewerten. Der Student benutzt eine Vielzahl von Datenbankzugriffsschnittstellen mit unterschiedlichem Abstraktionsniveau bei Programmierübungen. Er ist in der Lage, die Vor- und Nachteile von unterschiedlichen Zugriffsschnittstellen bzw. Datenbankmodellen einzuschätzen. Mit diesem gewonnenen Wissen wird der Student befähigt, bei der Entwicklung eines datenbankbasierten Informationssystems eine geeignete Systemarchitektur zu entwerfen und die Anforderungen der jeweiligen Anwendung zu berücksichtigen. Schwerpunktmaßig wird dieses Wissen auf die Entwicklung von Datenbanken im Web angewendet.

<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Der Student beherrscht einen Datenbankentwurf und kann einfache Anfragen mittels SQL formulieren.
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- M. Skulschus, M. Wiederstein: "Oracle, PL/SQL und XML", Comelio Medien, in der aktuellen Auflage.</li> <li>- H. Wehr, B. Müller: "Java Persistence API 2: Hibernate, EclipseLink, OpenJPA und Erweiterungen", Carl Hanser Verlag, 2012.</li> <li>- P. J. Sadalage, M. Fowler: „NoSQL Distilled“, Addison-Wesley, 2009.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul Informatik   Bachelor (20INB) Wahlpflicht Medieninformatik   Bachelor (20MIB) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Multimediale Webprogrammierung Multimedia Web Programming
<b>Modulnummer</b>	C741 Version: 3
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. habil. Michael Frank <a href="mailto:michael.frank@htwk-leipzig.de">michael.frank@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. habil. Michael Frank <a href="mailto:michael.frank@htwk-leipzig.de">michael.frank@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Reine Präsenzveranstaltung mit E-Learning-Inhalten
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- HTML5 und seine Strukturelemente, Dokumentstrukturierung, semantische Aspekte</li> <li>- CSS3: Flex- und Grid-Boxendesign, Bootstrap 4, Schatten, Farbverläufe, Transparenzen, Transformationen, SVG-Nutzung, SASS/SCSS</li> <li>- Nutzung von JavaScript und von JavaScript-Bibliotheken wie jQuery</li> <li>- HTML5-APIs wie Canvas, Drag&amp;Drop, Geolocation, Storage, File, Web Message, Web Worker, Mediaelement, WebRTC u.a..</li> <li>- Weitere Aspekte je nach Entwicklungen rund um HTML5.</li> <li>- Praktische Übungen aller Aspekte.</li> </ul>
<b>Qualifikationsziele</b>	<p>Die Studierenden beherrschen moderne Cross-Plattform-Webprogrammierung mit HTML5, CSS3, Web APIs und JavaScript-Bibliotheken unter Berücksichtigung von Aspekten unterschiedlicher Webbrowers. Die grundlegenden Modelle und Methoden der klassischen Webprogrammierung sind erlernt worden.</p> <p>Sie sind mit Prinzipien der Barrierefreiheit in der Webprogrammierung vertraut und befähigt, sich mit der weiteren dynamischen Entwicklung der Webprogrammierung selbstständig auseinanderzusetzen.</p>
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Kompetenzen in statischer Webprogrammierung mit HTML, CSS und JavaScript einschließlich DOM
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- J. D. Gauchat: „HTML5, CSS3 und JavaScript“, Wiley-VCH und Selbstverlag, 2013/2014.</li> <li>- J. Wolf: „HTML5 und CSS3: Das umfassende Handbuch zum Lernen und Nachschlagen. Inkl. JavaScript, Bootstrap, Responsive Webdesign u.v.m.“, Rheinwerk Computing, 2019.</li> <li>- St. Elter: „Programmieren lernen mit JavaScript“, Rheinwerk Computing, 2017.</li> <li>- J. Lett: „Bootstrap 4 Quick Start“, Bootstrap Creative, 2018.</li> <li>- F. Bongers, M. Vollendorf: „jQuery 3: Das umfassende Handbuch zum JavaScript-Framework“, Galileo Press, 2017.</li> <li>- Div. Schriftquellen und Internetquellen je nach Thematik und Zeitraum.</li> </ul>

<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Prüfungsvorleistung Belege (PVB): Übungsfragen und praktische Übungsaufgaben
<b>Verwendbarkeit</b>	<p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Medieninformatik   Bachelor Studienrichtung Bibliotheksinformatik (20MIB-BI) Pflichtmodul</p> <p>Informatik   Bachelor (20INB) Wahlpflicht</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Digitale Signal- und Bildverarbeitung Digital Signal and Image Processing
<b>Modulnummer</b>	C763 Version: 1
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Mario Hlawitschka <a href="mailto:mario.hlawitschka@htwk-leipzig.de">mario.hlawitschka@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	keine Angabe
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Signale, Zufallssignale und zeitdiskrete Zufallsprozesse</li> <li>- Abtastung zeitkontinuierlicher Signale</li> <li>- Lineare zeitinvariante Systeme</li> <li>- Diskrete Fourier Transformation</li> <li>- Analyse und Entwurf digitaler Filter</li> <li>- Anwendungen in der Medientechnik</li> <li>- Grundlagen der digitalen Bildverarbeitung</li> </ul> <p>Praktische Übungen mit der Messwerterfassungs- und Messwertverarbeitungssoftware LabVIEW und mit der Bildbearbeitungssoftware Adobe Photoshop</p>
<b>Qualifikationsziele</b>	Die Studierenden besitzen grundlegende Kenntnisse auf dem Gebiet der digitalen Signalverarbeitung, des Entwurfs digitaler Filter und der Anwendung von Signalverarbeitungsverfahren in der Bildverarbeitung und Medientechnik. Die praktische Fähigkeiten und Fertigkeiten zur Gestaltung von Signalverarbeitungsprozessen, zur Simulation von Signalverarbeitungsverfahren mittels LabVIEW, zum Filterentwurf und in der digitalen Bildverarbeitung wurden geschult.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	keine

<b>Literaturhinweise</b>	<p>A. V. Oppenheim, R. W. Schafer, J. R. Buck: "Zeitdiskrete Signalverarbeitung", Pearson Studium, 2004.</p> <p>M. Meyer: "Signalverarbeitung: Analoge und digitale Signale, Systeme und Filter", 6. Aufl., Vieweg+Teubner, 2011.</p> <p>K.-D. Kammeyer et al.: "Digitale Signalverarbeitung: Filterung und Spektralanalyse mit MATLAB®-Übungen", 8. Aufl., Vieweg+Teubner, 2012.</p> <p>M. Werner: "Digitale Signalverarbeitung mit MATLAB®: Grundkurs mit 16 ausführlichen Versuchen", 5. Aufl., Vieweg+Teubner, 2012.</p> <p>D. C. von Grünigen: "Digitale Signalverarbeitung: mit einer Einführung in die kontinuierlichen Signale und Systeme", 3. Aufl., Carl Hanser, 2004.</p>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	<p>Arbeitsaufwand: Projektarbeit (Referate) 30 h</p> <p>Prüfungsvorleistung Belege (PVB): Praktikumsaufgaben</p>
<b>Verwendbarkeit</b>	<p>Medieninformatik   Bachelor (20MIB) Pflichtmodul</p> <p>Informatik   Bachelor (20INB) Wahlpflicht</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Web Engineering Web Engineering
<b>Modulnummer</b>	C864 Version: 0
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Nach Bekanntgabe der Fakultät
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Andreas Both
<b>Dozierende</b>	Prof. Dr. rer. nat. Andreas Both
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Seminar)
<b>Selbststudienzeit</b>	90 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Testat
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Vorlesung, Seminar-Übungen
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	keine Angabe
<b>Qualifikationsziele</b>	Die Studierenden sind zur Konzeption und zur Entwicklung von webbasierten, Full-Stack-Anwendungslösungen befähigt. Sie beherrschen die aktuellen Konzeptionsparadigmen von Client/Server-Kommunikation sowie die Gestaltung einer nachhaltigen, sicheren, robusten, skalierende, komponentenorientierten Architektur.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	- Softwaretechnik (3. Fachsemester) - Datenbanken (3. Fachsemester)
<b>Literaturhinweise</b>	keine Angabe
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Dieses Modul folgt einem Full-Stack-Ansatz (also Frontend-, Backend- und Kommunikationstechnologien), um eine ganzheitliche Lösungsbetrachtung zu ermöglichen. Vorwissen in Web-Kommunikation (HTTP, Websockets) und Web-Technologien (HTML, CSS, JavaScript) ist hilfreich, aber nicht vorausgesetzt.
<b>Verwendbarkeit</b>	Wahlpflichtmodul: INB, Wahlpflichtmodul: MIB
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Formale Methoden und Werkzeuge Formal Methods and Tools
<b>Modulnummer</b>	C869 Version: 1
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Master
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Johannes Waldmann <a href="mailto:johannes.waldmann@htwk-leipzig.de">johannes.waldmann@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg Prüfungsvorleistung Präsentation
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Vorlesung mit Übung
<b>Medienform</b>	Vorlesungsfolien
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- <b>SAT-Solver und BDDs als aussagenlogische Werkzeuge</b></li> <li>- Beschreibung des Systemverhaltens durch aussagenlogische Formeln (bounded model checking)</li> <li>- Werkzeuge zur programmatischen Herstellung solcher Formeln (Bsp: ersatz)</li> <li>- Werkzeuge zu effizienten Bestimmung eines Modells (SAT-Solver, Bsp: kissat)</li> <li>- Werkzeuge zur Darstellung und effizienten Verarbeitung von Modellmengen als Boolesche Entscheidungsdiagramme (BDD),</li> <li>- <b>Model Checking mit endlichen Automaten</b></li> <li>- Beschreibung des Systemverhaltens durch eingeschränkte prädikatenlogische Formeln</li> <li>- effiziente Operationen auf endlichen Automaten, die Modellmengen darstellen</li> <li>- <b>Programmierwerkzeuge für präzise Typisierung</b></li> <li>- die Curry-Howard-Isomorphie (Typ = Aussage, Programm = Beweis)</li> <li>- Programmiersprachen mit dependent types (Bsp: Agda)</li> <li>- interaktive Beweis-Systeme (Bsp: Mimer für Agda)</li> </ul>
<b>Qualifikationsziele</b>	Wesentliche Methoden und Werkzeuge zum exakten Formulieren und Nachweisen von Eigenschaften von Softwaresystemen kennen, verstehen, auswählen, anwenden. Den Zusammenhang zu den zugrundeliegenden fundamentalen Ideen verstehen, die bereits aus anderen Vorlesungen bekannt sind (VL Modellierung: Logik, VL Automaten und formale Sprachen: Automaten, VL (Fortgeschrittene) Programmierung: Termersetzung, Lambda-Kalkül, Typsysteme).
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	regelmäßiges und erfolgreiches Bearbeiten von Übungsaufgaben

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- Cerone, A., Roggenbach, M., Schlingloff, B.-H., Schneider, G., Shaikh, S.A.: Teaching formal methods for software engineering - ten principles, <i>Informatica Didactica</i>, ? 9, (2015).  <a href="https://www.informaticadidactica.de/uploads/Artikel/Schlinghoff2015/Schlinghoff2015.pdf">https://www.informaticadidactica.de/uploads/Artikel/Schlinghoff2015/Schlinghoff2015.pdf</a>  <a href="https://www.informaticadidactica.de/uploads/Artikel/Schlinghoff2015/Schlinghoff2015.pdf">(https://www.informaticadidactica.de/uploads/Artikel/Schlinghoff2015/Schlinghoff2015.pdf)</a></li> <li>- Uwe Schöning, Jacob Toran: Das Erfüllbarkeitsproblem SAT, Lehmanns (2012)</li> <li>- Christel Baier, Joost Katoen: Principles of Model Checking. MIT Press, Cambridge (2008)</li> <li>- Samuel Mimram: Program = Proof,  <a href="https://www.lix.polytechnique.fr/Labo/Samuel.Mimram/teaching/INF551/course.pdf">https://www.lix.polytechnique.fr/Labo/Samuel.Mimram/teaching/INF551/course.pdf</a>  <a href="https://www.lix.polytechnique.fr/Labo/Samuel.Mimram/teaching/INF551/course.pdf">(https://www.lix.polytechnique.fr/Labo/Samuel.Mimram/teaching/INF551/course.pdf)</a></li> </ul>
<b>Aktuelle Lehrressourcen</b>	<p>Software (Auswahl):</p> <ul style="list-style-type: none"> <li>- Edward Kmett et al.: ersatz, A monad for expressing SAT or QSAT problems using observable sharing, 2010–, <a href="https://hackage.haskell.org/package/ersatz">https://hackage.haskell.org/package/ersatz</a>  <a href="https://hackage.haskell.org/package/ersatz">(https://hackage.haskell.org/package/ersatz)</a></li> <li>- Johannes Waldmann: boumchak Bounded Model Checker, 2017–, <a href="https://git.imn.htwk-leipzig.de/waldmann/boumchak">https://git.imn.htwk-leipzig.de/waldmann/boumchak</a>  <a href="https://git.imn.htwk-leipzig.de/waldmann/boumchak">(https://git.imn.htwk-leipzig.de/waldmann/boumchak)</a></li> <li>- Armin Biere: The Kissat SAT Solver, 2022–, <a href="https://github.com/arminbiere/kissat">https://github.com/arminbiere/kissat</a>  <a href="https://github.com/arminbiere/kissat">(https://github.com/arminbiere/kissat)</a></li> <li>- Ulf Norell, Andreas Abel, et al.: Agda, 2005–, <a href="https://wiki.portal.chalmers.se/agda/">https://wiki.portal.chalmers.se/agda/</a>  <a href="https://wiki.portal.chalmers.se/agda/">(https://wiki.portal.chalmers.se/agda/)</a></li> </ul>
<b>Hinweise</b>	Prüfungsvorleistungen: regelmäßiges und erfolgreiches Bearbeiten und Präsentieren von Übungsaufgaben
<b>Verwendbarkeit</b>	Informatik   Master (INM) Wahlpflichtmodul Medieninformatik   Master (MIM) Wahlpflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Assemblerprogrammierung Assembler Programming
<b>Modulnummer</b>	C960 Version: 2
<b>Fakultät</b>	FIM-INF: Informatikstudiengänge - Fakultät Informatik und Medien
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Jens Wagner <a href="mailto:jens.wagner@htwk-leipzig.de">jens.wagner@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Seminar)
<b>Selbststudienzeit</b>	94 Stunden 60 Stunden Bearbeitung Prüfungsleistung 34 Stunden Vorbereitung Lehrveranstaltung
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Projektarbeit Prüfungsdauer: 6 Wochen   Wichtung: 100%
<b>Lehr- und Lernformen</b>	keine Angabe
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- i486-Programmiermodell im Real address mode</li> <li>- Adressierungsarten</li> <li>- Einsatz verschiedener Assemblerbefehle</li> <li>- Unterprogramme, Parameterübergabetechniken</li> <li>- Interrupt-Verarbeitung</li> <li>- Gleitpunkt-Einheit</li> <li>- Assemblerprogrammierung mit einem ARM-Prozessor</li> </ul>
<b>Qualifikationsziele</b>	Die Studenten sollen die Möglichkeiten kennen und beherrschen, Programme durch Ausnutzung der Prozessorarchitektur zu optimieren. Die Studenten können mit typischen Problemen bei der hardwarenahen Programmierung umgehen.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Es werden Grundkenntnisse in der Programmierung und Rechnerarchitektur vorausgesetzt
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- J. Erdweg: „Assemblerprogrammierung mit dem PC“, Vieweg, 1992.</li> <li>- T. E. Podschun: „Das Assemblerbuch“, Addison-Wesley, 2002.</li> <li>- L. Pyeatt: Modern Assembly Language Programming with the ARM Processor, Elsevier 2016</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor Wahlpflichtmodul
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Einführung in ERP-Software (SAP) Introduction to ERP Software (SAP)
<b>Modulnummer</b>	N450 Version: 2
<b>Fakultät</b>	MNZ-Ma: Mathematik - Mathematisch-Naturwissenschaftliches Zentrum
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Tobias Martin <a href="mailto:tobias.martin@htwk-leipzig.de">tobias.martin@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Tobias Martin <a href="mailto:tobias.martin@htwk-leipzig.de">tobias.martin@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung am Computer Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Vorlesung mit integrierten Übungen
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Einführung in SAP Software</li> <li>- Navigation</li> <li>- Einführung in GBI</li> <li>- Vertrieb</li> <li>- Materialwirtschaft</li> <li>- Produktionsplanung und -steuerung</li> <li>- Finanzwesen</li> <li>- Controlling</li> <li>- Human Capital Management</li> <li>- Warehouse Management</li> <li>- Projektmanagement</li> <li>- Integrierte Fallstudien</li> </ul>
<b>Qualifikationsziele</b>	Die Studierenden können in SAP ERP Software navigieren, Transaktionen aufrufen und buchen. Sie können betriebliche Daten durch Reports in SAP ERP Software analysieren. Sie haben das Integrationsmodell verstanden und können integrierte Fallstudien in SAP ERP Software bearbeiten.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Kenntnisse Grundlagen der Betriebswirtschaft und Datenbanktechniken
<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- CDI (Hrsg.): „SAP R/3® Einführung“, Pearson, 2001.</li> <li>- A. Maassen et al.: „Grundkurs SAP R/3®: Lern- und Arbeitsbuch“, Vieweg, 2003.</li> <li>- P. Wenzel: „Betriebswirtschaftliche Anwendungen mit SAP R/3“, Vieweg+Teubner, 1999.</li> <li>- T. Teufel et al.: „SAP-Prozesse, Finanzwesen und Controlling“, Addison-Wesley, 2000.</li> <li>- F. Klenger, E. Falk-Kalms: „Kostenstellenrechnung mit SAP R/3“, Vieweg, 2002.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe

Verwendbarkeit	Informatik Bachelor (INB)
Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.	

<b>Modul</b>	Diskrete Mathematik und Optimierung Discrete Mathematics and Optimization
<b>Modulnummer</b>	N468 [N915] Version: 0
<b>Fakultät</b>	MNZ-Ma: Mathematik - Mathematisch-Naturwissenschaftliches Zentrum
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Sommersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. habil. Martin Grüttmüller <a href="mailto:martin.gruettmueller@htwk-leipzig.de">martin.gruettmueller@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. habil. Martin Grüttmüller <a href="mailto:martin.gruettmueller@htwk-leipzig.de">martin.gruettmueller@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 44 Stunden Vorbereitung Lehrveranstaltung 35 Stunden Sonstiges 15 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Keine
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Vorlesung, Übung, Selbstlernaufgaben
<b>Medienform</b>	Beamer, Tafel
<b>Lehrinhalte/Gliederung</b>	<ol style="list-style-type: none"> <li>1. Logik</li> <li>2. Graphentheorie</li> <li>2.1 Grundlagen</li> <li>2.2 Bäume, Minimalgerüste, Matroide</li> <li>2.3 Kürzeste Wege in Graphen</li> <li>2.4 Maximum Matchings</li> <li>2.5 Flüsse in Netzwerken</li> <li>3. Algebraische Strukturen (Modulare Arithmetik) und Ordnungsstrukturen</li> <li>4. Verbände</li> </ol>
<b>Qualifikationsziele</b>	Nach erfolgreichem Abschluss des Moduls haben die Studierenden grundlegende Kenntnisse auf dem Gebiet diskreter mathematischer Strukturen erworben. Dazu gehört insbesondere das Erkennen und Klassifizieren von Algebraischen- und Ordnungsstrukturen. Die Studierenden können logische Argumentationen nachvollziehen und selber korrekt führen. Sie sind in der Lage Algorithmen zur Lösung von Aufgaben einzusetzen und selbstständig zu entwickeln. Darüber hinaus besitzen die Studierenden Grundkenntnisse der Graphentheorie und kennen Standardoptimierungsprobleme. Sie haben ihre Problemlösungs- und Kommunikationsfähigkeiten weiterentwickelt, indem sie geeignete Anwendungsprobleme in einem graphentheoretischen Kontext unter Optimierungsgesichtspunkten modellieren, Fallvergleiche mit Standardoptimierungsproblemen durchführen, angemessene Hilfsmittel zur Lösung einsetzen und dabei die Fachsprache verstehen und anwenden.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Modellierungskompetenzen, Kompetenzen aus den Modulen Mathematik für Informatiker I und II sowie aus Algorithmen und Datenstrukturen

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- M. Aigner: "Diskrete Mathematik", SpringerVieweg 2006 [ebook].</li> <li>- B. Korte, J. Vygen: „Kombinatorische Optimierung“, SpringerSpektrum, 2018 [ebook]</li> <li>- R. Diestel: "Graphentheorie", Springer Verlag, 2017.</li> <li>- V. Turan: "Algorithmische Graphentheorie", Oldenbourg Wissenschaftsverlag [ebook].</li> <li>- D. Jungnickel: "Graphen, Netzwerke und Algorithmen", BI-Wissenschaftsverlag, 1990.</li> <li>- D. Jungnickel: "Graphs, Networks und Algorithms", Springer, 2013.</li> <li>- Weitere aktuelle Literaturhinweise werden in den Lehrveranstaltungen gegeben.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	OPAL-Kurs Vorlesungsskript Belegaufgaben
<b>Hinweise</b>	Selbststudienzeit: 35 Stunden für Bearbeitung fakultativer Belege
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Wahlpflicht  Medieninformatik   Bachelor (20MIB) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	

<b>Modul</b>	Numerische Algorithmen Numerical Mathematics
<b>Modulnummer</b>	N840 Version: 0
<b>Fakultät</b>	MNZ-Ma: Mathematik - Mathematisch-Naturwissenschaftliches Zentrum
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. rer. nat. Patrick Kürschner <a href="mailto:patrick.kuerschner@htwk-leipzig.de">patrick.kuerschner@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. rer. nat. Patrick Kürschner <a href="mailto:patrick.kuerschner@htwk-leipzig.de">patrick.kuerschner@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Übung)
<b>Selbststudienzeit</b>	94 Stunden 40 Stunden Vorbereitung Lehrveranstaltung 14 Stunden Bearbeitung Prüfungsvorleistung 20 Stunden Selbststudium 20 Stunden Vorbereitung Prüfung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Beleg
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Prüfungsdauer: 120 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Vorlesung: Tafel und Beamer, ein Lückenskript wird bereitgestellt Übungen: Besprechung Übungsaufgaben, Durchführung numerischer Experimente, Programmierung numerischer Algorithmen am Rechner
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	- Iterationsverfahren zur Lösung linearer Gleichungssysteme - Eigenwertprobleme - Numerische Quadratur und Kubatur - Optimierungsverfahren - Numerische Methoden für Randwertprobleme gewöhnlicher Differenzialgleichungen. - Numerische Methoden zur Lösung partieller Differenzialgleichungen
<b>Qualifikationsziele</b>	Die Studenten erwerben vertiefte Kenntnisse der numerischen Mathematik. Sie lernen algorithmische Aspekte numerischer Verfahren kennen und wissen numerische Methoden problemorientiert einzusetzen. Sie beherrschen sowohl die Verwendung existierender Programmprojekte als auch die Umsetzung numerischer Verfahren. Dazu wird das open source System Octave eingesetzt, welches weitestgehend befehlkompatibel zu MATLAB ist.
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Mathematik für Informatiker I, II, III
<b>Literaturhinweise</b>	- Numerische Mathematik, Hans Rudolf Schwarz, Norbert Köckler, Vieweg_Teubner, 2011 - Numerische Mathematik, Michael Knorrenshild, Fachbuchverlag Leipzig, 2008 - Numerische Methoden, 2. Auflage: Thomas Huckle, Stefan Schneider, Springer 2006 - Nichtlineare Optimierung, Michael Ulbrich, Stefan Ulbrich, Springer 2021 - Praktische Mathematik mit MATLAB, Scilab und Octave, Frank Thuselt, Felix Paul Gennrich, Springer Spektrum 2013 - Numerical Methods in Engineering with MATLAB, Third Edition, Jaan Kiusalaas, Cambridge University Press 2018

<b>Aktuelle Lehrressourcen</b>	keine
<b>Hinweise</b>	Keine Angabe
<b>Verwendbarkeit</b>	Informatik   Bachelor (20INB) Wahlpflicht
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	<a href="https://bildungspotrait.sachsen.de/opal/auth/RepositoryEntry/36273225728">https://bildungspotrait.sachsen.de/opal/auth/RepositoryEntry/36273225728</a>

<b>Modul</b>	Einführung in die Betriebswirtschaftslehre Introduction to Business Administration
<b>Modulnummer</b>	W233 [W861] Version: 2
<b>Fakultät</b>	FWW: Fakultät Wirtschaftswissenschaft und Wirtschaftsingenieurwesen
<b>Niveau</b>	Bachelor
<b>Dauer</b>	1 Semester
<b>Turnus</b>	Wintersemester
<b>Modulverantwortliche</b>	Prof. Dr. oec. publ. Sabine Hüttinger <a href="mailto:sabine.huettinger@htwk-leipzig.de">sabine.huettinger@htwk-leipzig.de</a>
<b>Dozierende</b>	Prof. Dr. oec. publ. Sabine Hüttinger <a href="mailto:sabine.huettinger@htwk-leipzig.de">sabine.huettinger@htwk-leipzig.de</a>  Dipl.-Kauffrau Gisela Schwetzler <a href="mailto:gisela.schwetzler@htwk-leipzig.de">gisela.schwetzler@htwk-leipzig.de</a>
<b>Sprache(n)</b>	Deutsch
<b>ECTS-Leistungspunkte</b>	5 ECTS-Punkte
<b>Workload</b>	150 Stunden
<b>Lehrveranstaltungen</b>	4 SWS (2 SWS Vorlesung   2 SWS Seminar)
<b>Selbststudienzeit</b>	94 Stunden 30 Stunden Bearbeitung Prüfungsvorleistung 64 Stunden Vorbereitung Lehrveranstaltung
<b>Prüfungsvorleistung(en)</b>	Prüfungsvorleistung Referat
<b>Prüfungsleistung(en)</b>	Prüfung Klausurarbeit Modulprüfung   Prüfungsdauer: 90 Minuten   Wichtung: 100%
<b>Lehr- und Lernformen</b>	Die Lehrveranstaltung ist interaktiv aufgebaut mit zahlreichen Übungen und einem Studentenreferat.
<b>Medienform</b>	keine Angabe
<b>Lehrinhalte/Gliederung</b>	<ul style="list-style-type: none"> <li>- Unternehmen und Umwelt</li> <li>- Typologie</li> <li>- Rechnungswesen intern (Kostenrechnung) und extern (Jahresabschluss)</li> <li>- Existenzgründung mit Businessplan</li> <li>- Marketing</li> <li>- Steuern</li> <li>- Insolvenzverfahren</li> <li>- Investitionsrechnung</li> <li>- Finanzierung</li> <li>- Controlling</li> <li>- Führung</li> </ul>
<b>Qualifikationsziele</b>	<p>Ziel ist die Vermittlung von grundlegenden betriebswirtschaftlichen Kenntnissen und Fertigkeiten.</p> <p>Fach- und methodische Kompetenzen: Kennen betriebswirtschaftlicher Begriffe und Denkweisen, Verstehen wichtiger betriebswirtschaftlicher Zusammenhänge, kunden- und kostenorientiertes Denken am Arbeitsplatz. Die Einführung in die Betriebswirtschaftslehre ermöglicht den Informatikern eine interdisziplinäre Sicht, die sie in ihrer beruflichen Entwicklung auch im Hinblick auf Führungsaufgaben unterstützen wird.</p>
<b>Zulassungsvoraussetzung</b>	Keine
<b>Empfohlene Voraussetzungen</b>	Keine

<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>- J. Drukarczyk: „Finanzierung“, UTB, in der aktuellen Auflage.</li> <li>- H. Meffert: „Marketing“, Gabler, in der aktuellen Auflage.</li> <li>- U. Pape: "Grundlagen der Finanzierung und Investition", De Gruyter, in der aktuellen Auflage</li> <li>- Wöhe/Döring: "Einführung in die Allgemeine Betriebswirtschaftslehre", Vahlen, in der aktuellen Auflage.</li> </ul>
<b>Aktuelle Lehrressourcen</b>	keine Angaben
<b>Hinweise</b>	<p>Prüfungsvorleistung:</p> <p>Referat (PVR): Referat mit max. 4 Teilnehmern</p>
<b>Verwendbarkeit</b>	<p>Informatik   Bachelor (20INB) Wahlpflicht</p> <p>Medieninformatik   Bachelor (20MIB) Wahlpflicht</p>
<b>Link zu Kurs/Lernressourcen im OPAL/Moodle/etc.</b>	