Fachbereich
Informatik und Kommunikation

Westfälische Hochschule
Gelsenkirchen Bocholt Recklinghausen

Modulkatalog

STUDIENGANG

Informatik (Bachelor)

10. April 2024

Inhaltsverzeichnis

Inhaltsverzeichnis 2
Modulkatalog 5
Algorithmen und Datenstrukturen 6
Algorithmen und Datenstrukturen 8
Bachelor-Arbeit Informatik 10
Bachelorarbeit Informatik 12
Betriebssysteme 14
Bildverarbeitung 16
Betriebssysteme 18
Datenbanksysteme 20
Datenbanksysteme 22
Echtzeitsysteme 24
Einführung in die Programmierung 26
Einführung in die Programmierung 28
Grundlagen der Mathematik für Informatiker 30
Internet-Sprachen 32
Internet-Datenbanken 34
Internet-Protokolle 36
Internet-Sprachen 39
Kolloquium zur Bachelorarbeit Informatik 41
Kolloquium zur Bachelor-Arbeit Informatik 43
Logik und diskrete Strukturen 45
Logik und diskrete Strukturen 47
Mathematik für Informatiker 49
Mensch-Computer-Interaktion 51
Mensch-Computer-Interaktion 54
Mathematische Grundlagen 56
Objektorientierte Programmierung 58
Objektorientierte Programmierung 60
Praxisphase 62
Prozedurale Programmierung 64
Praxisphase 66
Rechnernetze 68
Rechnernetze 70
Robotik 72
Statistik und Lineare Algebra 74
Softwareprojekt Informatik 76

Softwaretechnik 80
Softwareprojekt Informatik 82
Softwaretechnik 86
Technische Grundlagen der Informatik 88
Technisches Englisch für Informatiker 90
Technisches Englisch für Informatiker 92
Technische Grundlagen der Informatik 94
Theoretische Informatik 96
Theoretische Informatik 98
Wahlpflichtkatalog Praktische Informatik 100
Betrieb komplexer verteilter Systeme 100
Betriebswirtschaftslehre für Informatiker 102
Bildverarbeitung 104
Grundlagen der IT-Sicherheit 106
IT-Recht 108
Komponentenbasierte Softwareentwicklung 110
Künstliche Intelligenz 113
Mobile Computing 115
Mobile Robotik 117
Objektorientierte Programmierung mit C++ 119
Parallele Programmierung 121
Practical Security Attacks and Exploitation 123
Robotik 125
Software Design 127
Wahlpflichtkatalog Technische Informatik 129
Betriebswirtschaftslehre für Informatiker 129
Grundlagen der IT-Sicherheit 131
IT-Recht 133
Medientechnik 135
Mobile Robotik 138
Modellbasierter Entwurf von Regelsystemen 140
Objektorientierte Programmierung mit C++ 143
Parallele Programmierung 145
Practical Security Attacks and Exploitation 147
Systemtheorie 149
Zeitdiskrete Regelsysteme 151
Wahlpflichtkatalog Informatik 154
Betrieb komplexer verteilter Systeme 154

Einführung in die Bildverarbeitung 156
Data on the Web 158
Data Science in Practice 160
Einführung in die medizinische Informatik 162
Einführung in die Robotik 164
Internet-Protokolle 166
IT-Recht 169
Grundlagen der IT-Sicherheit 171
Komponentenbasierte Softwareentwicklung 173
Künstliche Intelligenz 176
Knowledge Graphs 178
Mobile und Cloud Computing 180
Mobile Robotik 182
Parallele Programmierung 184
Prozedurale Programmierung 186
Practical Security Attacks and Exploitation 188
Software Design 190

Modulkatalog

Algorithmen und Datenstrukturen

Kürzel:	ADS		
Untertitel:	---		
Studiensemester:	2. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr. Wolfram Conen		
Dozent(in):	Prof. Dr. Wolfram Conen		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	2	2	2
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung		
Gruppengröße:	Standard		
Arbeitsaufwand:	Kontaktzeit: 56 Zeitstunden Selbststudium: 124 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Sommersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Anmeldung über Moodle-Kurs zu diesem Modul		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	Logik und diskrete Strukturen, Einführung in die Programmierung		
Angestrebte Lernergebnisse:	Die Studierenden kennen wichtige grundlegende Resultate und Methoden der Algorithmik und können diese auf ausgewählte Problemstellungen anwenden. Sie gewinnen detaillierte Einblicke in die problemspezifische Optimierung von Algorithmen mittels geeignet gewählter Datenstrukturen und können diese nachvollziehen und anwenden. Sie kennen und beherrschen die Grundzüge der Analyse von Algorithmen und Problemen.		
Inhalt:	Wichtige Grundprobleme der Informatik und ihre Lösung mit Algorithmen und unterstützenden Datenstrukturen unter Berücksichtigung des Aufwandes, u.a.: Sortieren (Quick/Heap/Bucketsort; Buckets, Priority-Queues) Problemlösung mittels Suche (Baumstrukturen,Tiefen-, Breitensuche, iterative Deepening, BestFirst, A*) Zugriffsstrukturen (Indices, Hashing) Greedy-Algorithmen (Kruskal, Huffman-Codierung, Fractional Knapsack)		

	Grenzen der praktischen Lösbarkeit (Komplexität) von Problemen am Beispiel von Wegeproblemen: Algorithmik (Dijkstra-Varianten, MST) und Approximation (TSP/MST) Querschnittsthema: Analyse von Algorithmen (Kosten, Optimalität, Approximierbarkeit).
Studien- / Prüfungsleistungen:	Klausur (75 Min)
Literatur:	Skript, ergänzend: • Cormen, Leierson, Rivest, Stein: Introduction to Algorithms, MIT Press • Skiena: Algorithm Design Manual, Springer jeweils in aktueller Auflage.
Bemerkungen:	---

Algorithmen und Datenstrukturen

Kürzel:	ADS
Untertitel:	---
Studiensemester:	2. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Wolfram Conen
Dozent(in):	Prof. Dr. Wolfram Conen
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 2 I/TI: 2 MI: 2 WI: 2
Lehrform / SWS:	3 SWS Vorlesung, 2 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Logik und diskrete Strukturen, Einführung in die Programmierung
Angestrebte Lernergebnisse:	Die Studierenden kennen wichtige grundlegende Resultate und Methoden der Algorithmik und können diese auf ausgewählte Problemstellungen anwenden. Sie gewinnen detaillierte Einblicke in die problemspezifische Optimierung von Algorithmen mittels geeignet gewählter Datenstrukturen und können diese nachvollziehen und anwenden. Sie kennen und beherrschen die Grundzüge der Analyse von Algorithmen und Problemen.
Inhalt:	Wichtige Grundprobleme der Informatik und ihre Lösung mit Algorithmen und unterstützenden Datenstrukturen unter Berücksichtigung des Problemlöseaufwandes, u.a.: Sortieren (Quick/Heap/Bucketsort; Buckets, Priority-Queues), Problemlösung mittels Suche (Tiefen-, Breitensuche, Iterative Deepening, BestFirst, A*);

	Zugriffsstrukturen (Hashing), Greedy-Algorithmen (Kruskal, Huffman-Codierung, Fractional Knapsack); Grenzen der praktischen Lösbarkeit (Komplexität) von Problemen am Beispiel von Wegeproblemen: Algorithmik (Dijkstra-Varianten, MST) und Approximation (TSP/MST), Analyse von Algorithmen (Kosten, Optimalität, Approximierbarkeit) Aufwand: Wichtige Probleme (80 %), Komplexität (20%)
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Klausur (75 Min.)
Literatur:	Skript, ergänzend: Cormen, Leierson, Rivest, Stein: Introduction to Algorithms, MIT Press, Owsnicki-Klewe: Algorithmen und Datenstrukturen, Wißner, Güting, Dieker: Datenstrukturen und Algorithmen, Teubner, jeweils in aktueller Auflage.
Bemerkungen:	---

Bachelor-Arbeit Informatik

Kürzel:	BAIN			
Untertitel:	Abschlussarbeit des Bachelor-Studiums der Informatik			
Studiensemester:	6. (Bachelor)			
Modulverantwortliche(r):	Studiengangsbeauftragte/r Informatik			
Dozent(in):	Alle Professoren der Fachgruppe Informatik			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	6	6	-	-
Lehrform / SWS:	Bachelor-Arbeit			
Gruppengröße:	Siehe § 22 der Rahmenprüfungsordnung			
Arbeitsaufwand:	360 Zeitstunden			
Leistungspunkte:	12			
Turnus:	Die Vergabe einer Bachelor-Arbeit ist jederzeit möglich.			
Teilnehmerzahl:	Wie Gruppengröße			
Anmeldungsmodalitäten:	Siehe § 23 und § 24 der Rahmenprüfungsordnung			
Voraussetzungen nach Prüfungsordnung:	Siehe § 23 der Rahmenprüfungsordnung			
Empfohlene Voraussetzungen (Modulprüfungen):	150 Leistungspunkte			
Angestrebte Lernergebnisse:	Die/der Studierende ist in der Lage, innerhalb einer vorgegebenen Frist eine praxisorientierte Aufgabe aus der praktischen oder technischen Informatik sowohl in ihren fachlichen Einzelheiten als auch in ihren themen- und fachübergreifenden Zusammenhängen nach wissenschaftlichen und fachpraktischen Methoden selbstständig zu bearbeiten und zu dokumentieren.			
Inhalt:	Es wird ein in der Regel praxisorientiertes Problem aus der praktischen oder technischen Informatik mit den im Studium erlernten Konzepten, Verfahren und Methoden in begrenzter Zeit unter Anleitung eines erfahrenen Betreuers gelöst.			
Studien- / Prüfungsleistungen:	Siehe § 24 und § 25 der Rahmenprüfungsordnung			
Literatur:	Franck, N.; Stary, J.: Die Technik wissenschaftlichen Arbeitens. UTB-Verlag Stuttgart, 2013 (17. überarb. Auflage), 301 Seiten, ISBN: 978-3825240400 Karmasin, M; Ribing, R.: Die Gestaltung wissenschaftlicher Arbeiten: Ein Leitfaden für Seminararbeiten, Bachelor-, Master- und			

	Magisterarbeiten sowie Dissertationen. UTB-Verlag Stuttgart, 2014 (8. aktual. Auflage), 167 Seiten, ISBN: 978-3825242596 Weitere themenspezifische Literatur
Bemerkungen:	---

Bachelorarbeit Informatik

Kürzel:	BAIN		
Untertitel:	Abschlussarbeit des Bachelor-Studiums der Informatik		
Studiensemester:	6. (Bachelor)		
Modulverantwortliche(r):	Studiengangsbeauftragte/r Informatik		
Dozent(in):	Alle Professorinnen und Professoren der Fachgruppe Informatik		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	6	-	-
Lehrform / SWS:	Bachelorarbeit		
Gruppengröße:	Siehe § 22 BRPO		
Arbeitsaufwand:	360 Zeitstunden		
Leistungspunkte:	12		
Turnus:	Die Vergabe einer Bachelorarbeit ist jederzeit möglich		
Teilnehmerzahl:	Wie Gruppengröße		
Anmeldungsmodalitäten:	Siehe § 24 BRPO		
Voraussetzungen nach Prüfungsordnung:	Siehe § 16 PO und § 23 BRPO		
Empfohlene Voraussetzungen (Modulprüfungen):	150 Leistungspunkte		
Angestrebte Lernergebnisse:	Die/der Studierende ist in der Lage, innerhalb einer vorgegebenen Frist eine praxisorientierte Aufgabe aus der praktischen Informatik sowohl in ihren fachlichen Einzelheiten als auch in ihren themen- und fachübergreifenden Zusammenhängen nach wissenschaftlichen und fachpraktischen Methoden selbstständig zu bearbeiten und zu dokumentieren.		
Inhalt:	Es wird ein in der Regel praxisorientiertes Problem aus der praktischen Informatik mit den im Studium erlernten Konzepten, Verfahren und Methoden in begrenzter Zeit unter Anleitung eines erfahrenen Betreuers gelöst.		
Studien- / Prüfungsleistungen:	Siehe § 24 und § 25 BRPO und siehe § 18 PO		
Literatur:	Franck, N.; Stary, J.: Die Technik wissenschaftlichen Arbeitens. UTB-Verlag Stuttgart, 2013 (17. überarb. Auflage), 301 Seiten, ISBN: 978-3825240400 Karmasin, M; Ribing, R.: Die Gestaltung wissenschaftlicher Arbeiten: Ein Leitfaden für Seminararbeiten, Bachelor-, Master- und		

	Magisterarbeiten sowie Dissertationen. UTB-Verlag Stuttgart, 2014 (8. aktual. Auflage), 167 Seiten, ISBN: 978-3825242596 Weitere themenspezifische Literatur
Bemerkungen:	-

Betriebssysteme

Kürzel:	BSY
Untertitel:	Einführung in die technologischen und organisatorischen Konzepte von modernen Betriebssystemen
Studiensemester:	3. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Andreas Cramer
Dozent(in):	Prof. Dr. Andreas Cramer
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 3 I/TI: 3 MI: 3 WI: -
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 58 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 122 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Technische Grundlagen der Informatik
Angestrebte Lernergebnisse:	Die Studierenden lernen die grundlegenden Konzepte und Verfahren von Betriebssystemen kennen. Sie erlangen die Fähigkeit, neue Betriebssystemkonzepte schnell begreifen, einordnen und bewerten zu können.
Inhalt:	• Einführung • Prozesse • Speicherverwaltung • Dateisystem • Ein-/Ausgabe • Unix
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: keine Prüfungsleistungen: Klausur

Literatur:	Andrew S. Tanenbaum: Moderne Betriebssysteme. 3. aktualisierte Auflage, ISBN: 978-3-8273-7342-7, Pearson Studium, 2009. Eduard Glatz: Betriebssysteme. Grundlagen, Konzepte, Systemprogrammierung. 3. Überarbeitete und aktualisierte Auflage ISBN: 3864902223, dpunkt.verlag 2015. Andrew S. Tanenbaum, Herbert Bos: Modern Operating Systems. 4. Edition, ISBN: 1292061421, Pearson Global Edition 2014.
Bemerkungen:	---

Bildverarbeitung

Kürzel:	BV
Untertitel:	Grundlegende Verfahren und Algorithmen der digitalen Bildverarbeitung
Studiensemester:	5. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Hartmut Surmann
Dozent(in):	Prof. Dr. Hartmut Surmann
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: 5; MI: WP; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	keine
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung, Grundlagen der Mathematik für Informatiker, Lineare Algebra
Angestrebte Lernergebnisse:	Die Studierenden kennen die Begriffe und Verfahren der digitalen Bildverarbeitung sowie die Konzepte und Methoden der Programmierung und können diese effektiv und strukturiert bei der Entwicklung eigener Bildverarbeitungsprogramme einsetzen. Lehrsprachen sind C / C++, Python. Bibliotheken: OpenCV, scikit-image
Inhalt:	• Grundlagen / Begriffsbildung • Kameras • Bildverarbeitungsoperationen • Morphologische Operationen • Bildsegmentierung

	• Merkmale von Objekten • Klassifikation • Bildverarbeitungssoftware
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• J. Steinmüller: „Bildanalyse“, Springer Verlag, ISBN 978-3-540-79743-2. • A. Nischwitz, P. Haberäcker: „Computergrafik und Bildverarbeitung, Band II Bildverarbeitung“, Teubner Verlag, ISBN 978-3-834-81712-9. • A Kaehler, G. Bradski: „Computer Vision in C++ with the OpenCV Library“, O'Reilly, ISBN 978-1-449-31465-1.
Bemerkungen:	---

Betriebssysteme

Kürzel:	BSY
Untertitel:	Einführung in die technologischen und organisatorischen Konzepte von modernen Betriebssystemen
Studiensemester:	2. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Andreas Cramer
Dozent(in):	Prof. Dr. Andreas Cramer
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: 2 ID: - WI: -
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Vorlesung nicht begrenzt, Übung: 30
Arbeitsaufwand:	Kontaktzeit: 56 Zeitstunden Selbststudium: 124 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Technische Grundlagen der Informatik
Angestrebte Lernergebnisse:	Die Studierenden lernen die grundlegenden Konzepte und Verfahren von Betriebssystemen kennen. Sie erlangen die Fähigkeit, neue Betriebssystemkonzepte schnell begreifen, einordnen und bewerten zu können.
Inhalt:	Einführung in Betriebssysteme Prozesse Speicherverwaltung Dateisystem Ein-/Ausgabe Unix
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: keine Prüfungsleistung: Klausur
Literatur:	Bekanntgabe in der Vorlesung

Bemerkungen:	---

Datenbanksysteme

Kürzel:	DBA
Untertitel:	---
Studiensemester:	3. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Katja Zeume
Dozent(in):	Prof. Dr. Katja Zeume
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 3 I/TI: 3 MI: 3 WI: 3
Lehrform / SWS:	3 SWS Vorlesung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 72 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 108 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung
Angestrebte Lernergebnisse:	Die Studierenden kennen die Grundlagen von Datenbanksystemen und deren Einsatz in der Praxis. Die Studierenden kennen die wesentliche Vorgehensweise und Methoden, um Realweltausschnitte zu modellieren und in gut strukturierte Datenbankschemata zu überführen. Die Studierenden sind in der Lage, Informationssysteme unter Einsatz von Datenbankprogrammierschnittstellen und der Datenbanksprache SQL zu entwickeln und zu optimieren.
Inhalt:	Die Veranstaltung bietet einen Einstieg in Datenbanksysteme und deren Anwendungen in der Praxis. Der Inhalt der **Vorlesungen und Praktika** ist wie folgt strukturiert: • Einführung in Datenbanksysteme

	• Anwendungsfälle von Datenbanksystemen in der Praxis • Das Datenbankmanagementsystem und seine Komponenten • Datenbankschemata und Konsistenzbedingungen • Relationale Algebra • Grundlagen SQL und SQL-Optimierung • (Optional) XML • (Optional) Ausblick auf nicht-relationale und NOSQL Datenbanken Das **Praktikum** enthält praktische Übungen zum Datenbankschemadesign und der Anwendung von SQL
Studien- / Prüfungsleistungen:	Prüfungsleistung: Klausur (75 Min.)
Literatur:	Heuer, Sattler, Saake. Datenbanken: Konzepte und Sprachen. mitp-Verlag Elmasri, Navathe. Grundlagen von Datenbanksystemen. Pearson Studium Foundations of Databases, Serge Abiteboul, Rick Hull, Victor Vianu, 1995. Ramakrishnan, Gehrke. Database Management Systems. McGraw-Hill
Bemerkungen:	---

Datenbanksysteme

Kürzel:	DBA		
Untertitel:	-		
Studiensemester:	3. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr. Katja Zeume		
Dozent(in):	Prof. Dr. Katja Zeume		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	3	3	3
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum		
Gruppengröße:	Vorlesung: nicht begrenzt, Praktikum: 20		
Arbeitsaufwand:	Kontaktzeit: 75 Zeitstunden Selbststudium: 105 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Wintersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	Grundlagen Programmierung		
Angestrebte Lernergebnisse:	Die Studierenden kennen die Grundlagen von Datenbanksystemen und deren Einsatz in der Praxis. Die Studierenden kennen die wesentliche Vorgehensweise und Methoden, um Realweltausschnitte zu modellieren und in gut strukturierte Datenbankschemata zu überführen. Die Studierenden sind in der Lage, Informationssysteme unter Einsatz von Datenbankprogrammierschnittstellen und der Datenbanksprache SQL zu entwickeln und zu optimieren.		
Inhalt:	Die Veranstaltung bietet einen Einstieg in Datenbanksysteme und deren Anwendungen in der Praxis. Der Inhalt der **Vorlesungen, Übungen und Praktika** ist wie folgt strukturiert: • Einführung in Datenbanksysteme • Anwendungsfälle von Datenbanksystemen in der Praxis		

	• Das Datenbankmanagementsystem und seine Komponenten • Datenbankschemata und Konsistenzbedingungen • Relationale Algebra • Grundlagen SQL und SQL-Optimierung • (Optional) XML • (Optional) Ausblick auf nicht-relationale und NOSQL Datenbanken **Übungen und Praktikum** enthalten praktische Aufgaben zum Datenbankdesign und der Anwendung von SQL.
Studien- / Prüfungsleistungen:	Prüfungsleistung: Klausur (75min)
Literatur:	• Heuer, Sattler, Saake. Datenbanken: Konzepte und Sprachen. mitp-Verlag • Elmasri, Navathe. Grundlagen von Datenbanksystemen. Pearson Studium • Foundations of Databases, Serge Abiteboul, Rick Hull, Victor Vianu, 1995. • Ramakrishnan, Gehrke. Database Management Systems. McGraw-Hill
Bemerkungen:	-

Echtzeitsysteme

Kürzel:	EZS
Untertitel:	---
Studiensemester:	4. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Ekkehard Schrey
Dozent(in):	Prof. Dr. Ekkehard Schrey
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: - I/TI: 4 MI: - WI: -
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Technische Grundlagen der Informatik, Logik & Diskrete Strukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung, Betriebssysteme
Angestrebte Lernergebnisse:	Die Studierenden • erlangen Kenntnisse über Aufbau, Funktionsweise und die Besonderheiten von (eingebetteten) Echtzeitsystemen und sind in der Lage, die Kenntnisse bei der Lösung konkreter Aufgabenstellungen selbständig anzuwenden. • erlangen Kenntnisse über die Besonderheiten der Programmierung von Echtzeitsystemen und sind in der Lage, die Kenntnisse bei konkreten Aufgabenstellungen selbständig anzuwenden.

Inhalt:	• Einführung in Aufgabenstellung von Echtzeitsystemen • Anforderungen an Echtzeitsysteme • Rechnerarchitekturen für Echtzeitsysteme • Einführung Maschinenprogrammierung • Unterbrechungsverarbeitung auf Maschinenebene • Architekturmerkmale zur Beschleunigung des Programmablaufs • Peripheriebauelemente für Echtzeitsysteme • Algorithmen für Echtzeitsysteme • Programmierung von Echtzeitsystemen • Entwurf und Synthese von Echtzeitsystemen • Echtzeitbetriebssysteme • Sicherheit und Zuverlässigkeit von Echtzeitsystemen
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Vorrechnung einer gewissen Anzahl von Übungsaufgaben. Die Anzahl der Aufgaben wird zu Beginn der Veranstaltung so festgelegt, dass jeder Studierende diese Anzahl gut bewältigen kann. Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.), alternativ mdl. Prüfung (30 Minuten)
Literatur:	Tanenbaum, A. S.: Computerarchitektur; Prentice Hall, 2006, ISBN 3-8273-7151-1 Flik, T; Mikroprozessortechnik und Rechnerstrukturen, Springer, 2005, ISBN 3-540-22270-7 Wörn,H; Brinkschulte, U.: Echtzeitsysteme. Springer, 2005, ISBN-10 3-540-20588-8 Liu, J.W.S.: Real Time Systems, Prentice Hall, 2000 ISBN 0-13-099651-3
Bemerkungen:	---

Einführung in die Programmierung

Kürzel:	EPR
Untertitel:	Grundlagen und Prinzipien der Programmierung
Studiensemester:	1. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Marcel Luis
Dozent(in):	Prof. Dr. Marcel Luis
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 1; I/TI: 1; MI: 1; WI: 1
Lehrform / SWS:	3 SWS Vorlesung (davon 1 SWS seminaristisch), 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 72 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 138 Zeitstunden
Leistungspunkte:	7
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Keine
Angestrebte Lernergebnisse:	Die Studierenden kennen die Grundelemente der funktionalen, imperativen und objektorientierten Programmierung. Sie kennen den Begriff des Algorithmus und verschiedene Ansätze zum Entwurf von Algorithmen. Anhand von Beispielen gewinnen sie Verständnis für die Themen Effizienz und Korrektheit. Die Studierenden sind somit in der Lage, zu einfachen Aufgabenstellungen qualitativ gute Lösungen zu konzipieren und zu realisieren. Lehrsprache ist Java.
Inhalt:	Begriff des Algorithmus • Datentypen • Struktur, Repräsentation und Auswertung von Ausdrücken • Funktionen • Rekursion • Klassen und Objekte • Zustände • Kontrollstrukturen • Entwurfsansätze für Algorithmen • Felder • rekursive Datenstrukturen
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine

	Prüfungsleistungen: Klausur (120 Min.)
Literatur:	Heinz-Peter Gumm, Manfred Sommer: Einführung in die Informatik. Oldenbourg, 2013. Joachim Goll, Cornelia Heinisch: Java als erste Programmiersprache. Springer Vieweg, 2014. Guido Krüger, Heiko Hansen: Java-Programmierung -- Das Handbuch zu Java 8. O'Reilly, 2014. Christian Ullenboom: Java ist auch eine Insel. Rheinwerk Computing, 2014. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley: The Java Language Specification, Java SE 8 Edition. Oracle America, Inc., 2013.
Bemerkungen:	---

Einführung in die Programmierung

Kürzel:	EPR
Untertitel:	Grundlagen und Prinzipien der Programmierung
Studiensemester:	1. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Marcel Luis
Dozent(in):	Prof. Dr. Marcel Luis
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: 1 ID: 1 WI: 1
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 75 Zeitstunden Selbststudium: 105 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Keine
Angestrebte Lernergebnisse:	• Die Studierenden kennen und verstehen die grundlegenden Elemente der imperativen und objektorientierten (noch ohne Klassenhierarchie) Programmierung. • Sie können Rekursion und Iteration adäquat zur Realisierung wiederholender Abläufe einsetzen. • Anhand von Anwendungsbeispielen gewinnen sie ein grundlegendes Verständnis für die Themen Effizienz und Korrektheit. • Die Studierenden wissen, dass Dokumentation und Test untrennbar mit Programmierung verbunden sind. • Sie sind insgesamt in der Lage, zu einfachen Aufgabenstellungen qualitativ gute Lösungen (in der Lehrsprache Java) zu konzipieren und zu realisieren.
Inhalt:	Begriff des Algorithmus • elementare Datentypen • Typen und Werte von Ausdrücken • Rekursion und Strategien zur Entwicklung rekursiver Lösungen • Klassen und Objekte • statische und Instanzmethoden •

	Dokumentation von Klassen und Methoden • Kontrollstrukturen • Entwurfsansätze für iterative Lösungen • Kapselung und Abstraktion • Felder • rekursive Datenstrukturen
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.)
Literatur:	• Joachim Goll, Cornelia Heinisch: Java als erste Programmiersprache. Springer Vieweg, 2016. • Christian Ullenboom: Java ist auch eine Insel. Rheinwerk Computing, 2021. • Offizielle Spezifikation der jeweils aktuellen Java-Version als Nachschlagewerk
Bemerkungen:	---

Grundlagen der Mathematik für Informatiker

Kürzel:	GMI
Untertitel:	---
Studiensemester:	1. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Laura Anderle
Dozent(in):	Prof. Dr. Laura Anderle
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 1 I/TI: 1 MI: 1 WI: 1
Lehrform / SWS:	4 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 72 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 138 Zeitstunden
Leistungspunkte:	7
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	---
Angestrebte Lernergebnisse:	Sicherer, praktischer Umgang mit den grundlegenden Begriffen der Höheren Mathematik, den Eigenschaften reeller Funktionen und diskreter Zahlenfolgen. Erkennen der praktischen Anwendungsmöglichkeiten in der Informatik. Sicherer und praktischer Umgang mit den grundlegenden Begriffen des Differentialkalküls
Inhalt:	Elemente der Mengenlehre, Reelle Zahlen, Intervalle, Ungleichungen, Fakultäten, Binomialkoeffizienten, Binomischer Lehrsatz, Mathematische Beweisprinzipien, Diskrete Zahlenfolgen, Konvergenzbegriff, monotone Folgen, unendliche Reihen, Funktion und Relation, allgemeine Funktionseigenschaften, spezielle reelle Funktionen und ihre wesentlichen Eigenschaften, Ableitungsbegriff, Ableitungsregeln, Höhere Ableitungen, Taylorformel mit Restglied, Darstellung reeller Funktionen durch

	Taylorreihen, relative und absolute Extrema, Regeln von L'Hospital
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Klausur (90 Min.)
Literatur:	Nehrlich: Diskrete Mathematik / Basiswissen für Informatiker, Fachbuchverlag Leipzig, 2003 Heuser: Lehrbuch der Analysis, Teil 1, Teubner Verlag, 2001 Timmann: Repetitorium der Analysis, Teil 1, Binomi Verlag, 2000 Dörfler, Peschek : Einführung in die Mathematik für Informatiker, Hanser-Verlag, 1998,
Bemerkungen:	Erheblich erweiterte Literaturliste im Netz

Internet-Sprachen

Kürzel:	INS
Untertitel:	---
Studiensemester:	3. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Andreas Cramer
Dozent(in):	Prof. Dr. Andreas Cramer
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: 3; ID: -; WI: WP
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Bestandene Prüfung in „Einführung in die Programmierung“ (EPR)
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Objektorientierte Programmierung
Angestrebte Lernergebnisse:	Die Studierenden lernen unterschiedliche Beschreibungssprachen und deren Einsatzgebiete kennen und bekommen erste praktische Erfahrungen mit deren Anwendung. Die Studierenden erlernen Verfahren zur Erstellung dynamischer Web-Seiten und wenden das Erlernte im Praktikum an. Sie erlangen die Fähigkeit, neue Konzepte im Umfeld der Internet-Sprachen schnell begreifen, einordnen und bewerten zu können.
Inhalt:	• HTML • CSS • PHP • XML, Verarbeitung von XML-Dateien mit Java, XML-Schema, XSLT, … • JavaScript, AJAX • Web-Services

Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: keine Prüfungsleistung: Klausur
Literatur:	Bekanntgabe in der Vorlesung
Bemerkungen:	-

Internet-Datenbanken

Kürzel:	IDB
Untertitel:	---
Studiensemester:	4. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Katja Zeume
Dozent(in):	Prof. Dr. Katja Zeume
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 4 I/TI: - MI: - WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 54 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 128 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Datenbanksysteme, Internetsprachen
Angestrebte Lernergebnisse:	Die Studierenden verstehen die technologischen Grundlagen webbasierter Informationssysteme. Die Studierenden kennen die gängigen Anwendungsfelder der webbasierten Datenverarbeitung und die spezifischen Probleme, die im Einsatz auftauchen Die Studierenden sind in der Lage, webbasierte Informationssysteme unter Einsatz von Web-Frameworks produktiv zu erstellen.
Inhalt:	Die Veranstaltung bietet eine Vertiefung in aktuelle erweiterte Datenbankformate und Anfragesprachen im Kontext von webbasierten und Cloud-Anwendungen. • Konzept Objekt-Relationales Mapping und Umsetzung am Beispiel aktueller Framework Implementierungen

	• Bedeutung semi-strukturierte Formate (z.Bsp. XML oder JSON), sowie passenden Anfrage- und Schemasprachen. • Einführung unstrukturierte Datenbankformate (sog. NOSQL Datenbanken) am Beispiel Graph-Datenbanken. • (Wahlweise) Überblick über weitere Formate aus dem Bereich TextDB und StreamingDB. • (Wahlweise) Aspekte von Cloud Maturity für Datenbasierte Anwendungen im Web Die einzelnen Themen werden mit Anwendungsfällen aus der Praxis motiviert und auf ihre Vor- und Nachteile verglichen
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Klausur (75 Min.)
Literatur:	B. Müller, H. Wehr: Java Persistence API , Hanser Verlag, aktuelle Ausgabe M.Kurz, M.Marinschek: JavaServer Faces: Grundlagen und erweiterte Konzepte, dpunkt Verlag, aktuelle Ausgabe Leskovec, Rajaraman, Ullman. Mining of Massive Datasets
Bemerkungen:	---

Internet-Protokolle

Kürzel:	INP
Untertitel:	Grundlegende Kenntnisse über die Aufgaben, Prinzipien, Mechanismen und Architekturen auf den unterschiedlichen Kommunikationsebenen
Studiensemester:	4. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. (TU NN) Norbert Pohlmann
Dozent(in):	Prof. Dr. (TU NN) Norbert Pohlmann
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 4 I/TI: - MI: WP WI: -
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 54 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 126 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Keine
Angestrebte Lernergebnisse:	• Gutes Verständnis für die fundamentalen Kommunikationsarchitekturen und -protokolle des Internets. • Erlangen von Kenntnissen über die Aufgaben, Prinzipien, Mechanismen und Architekturen auf den unterschiedlichen Kommunikationsebenen. • Gewinnen von praktischen Erfahrungen über die Kommunikationsprotokolle, Kommunikationsdienste und -anwendungen durch Versuche und mit Hilfe von Protokollanalysen. • Erleben der Notwendigkeit und Wichtigkeit der Lehrinhalte.

Inhalt:	• Einführung: Begriffe, geschichtliche Entwicklung, Beispiele für Netzwerke, die Zukunft von Netzwerken und des Internets • Das ISO- und TCP/IP-Referenzmodell: Instanzen, Dienste, Protokolle, Paketstrukturen; Schichtenaufgaben • Netzkoppelelemente: Repeater, Hubs, Bridges, Switches, Router, Gateway • Vermittlungsebene: Aufgaben der Vermittlungsebene (IP, ARP, ICMP, Routingprotokolle); Begriffe/Mechanismen der Vermittlungstechnik (Warteschlangen, Routingverfahren, Traffic Shaping, Scheduling, Call admission control); Quality of Service in IP-Netzen (Idee, Konzept, IntServ, RSVP, DiffServ, MLPS) • Transportebene: Dienste und Mechanismen der Transportschicht (TCP, UDP; RTP); Sequenz- und Bestätigungsnummern, Prüfsumme, Zeitüberwachung, Segmentierung, Stream-Service, Sliding-Windows-Technik, Slow-Start, Congestion Windows, Delayed acknowledgement, Nagle Algorithmus • Anwendungsebene: DNS (Domain Name Service), SMTP (E- Mail), HTTP (World Wide Web), SIP (Session Initiation Protocol) Pro Anwendungsdienst: Kommandos, Nachrichten/Datentypen, Verbindungen/Kommunikation, Besonderheiten; Protokollanalysen und deren Bewertung • Client-Server- und P2P-Architektur • Struktur und Aufbau des Internets (AS, Arten von ASe, Verbindungen, CDN, …) • Grundlagen von Verteilten Systemen (Motivation, Ziele, Konzepte, Beispiele, …)
Studien- / Prüfungsleistungen:	Studienleistungen: Erfolgreich absolviertes Praktikum Prüfungsleistungen: Klausur (90 Min.)
Literatur:	• Tanenbaum, A.: "Computernetzwerke"; Prentice Hall, 2003; ISBN: 3- 8273-7046-9 • Tanenbaum, A.; van Stehen, M.: "Verteilte Systeme - Grundlagen und Paradigmen"; Prentice Hall, 2003; ISBN: 3-8273-7057-4 • Proebster, W: "Rechnernetze - Technik, Protokolle, Systeme, Anwendungen"; Oldenbourg Verlag; ISBN: 3-486-25777-3 • Müller, G.; Eymann, T.; Kreutzer, M.: "Telematik- und Kommunikationssysteme in der vernetzten

	Wirtschaft"; Oldenbourg Verlag; ISBN: 3-486-25888-5 • Wander, K.; "Protokolle und Dienste der Informationstechnologie"; Interest Verlag; ISBN: 3-8245-0412-X • S. Feld, N. Pohlmann, M. Sparenberg, B. Wichmann: „Analyzing G-20´Key Autonomous Systems and their Intermeshing using AS-Analyzer”. In Proceedings of the ISSE 2012 - Securing Electronic Business Processes - Highlights of the Information Security Solutions Europe 2012 Conference, Eds.: N. Pohlmann, H. Reimer, W. Schneider; Springer Vieweg Verlag, Wiesbaden 2012
Bemerkungen:	---

Internet-Sprachen

Kürzel:	INS
Untertitel:	---
Studiensemester:	3. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Andreas Cramer
Dozent(in):	Prof. Dr. Andreas Cramer
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 3 I/TI: - MI: 3 WI: WP
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 58 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 122 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Bestandene Prüfung in „Einführung in die Programmierung“ (EPR)
Empfohlene Voraussetzungen (Modulprüfungen):	Objektorientierte Programmierung
Angestrebte Lernergebnisse:	Die Studierenden lernen unterschiedliche Beschreibungssprachen und deren Einsatzgebiete kennen und bekommen erste praktische Erfahrungen mit deren Anwendung. Die Studierenden erlernen Verfahren zur Erstellung dynamischer Web-Seiten und wenden das Erlernte im Praktikum an. Sie erlangen die Fähigkeit, neue Konzepte im Umfeld der Internet-Sprachen schnell begreifen, einordnen und bewerten zu können.
Inhalt:	HTML CSS XML, Verarbeitung von XML-Daten mit Java, XML-Schema, XSLT, ... PHP JavaScript

	AJAX Web-Services
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur
Literatur:	Bekanntgabe in der Vorlesung
Bemerkungen:	---

Kolloquium zur Bachelorarbeit Informatik

Kürzel:	KBIN		
Untertitel:	Abschlussprüfung im Bachelor-Studium der Informatik		
Studiensemester:	6. (Bachelor)		
Modulverantwortliche(r):	Studiengangsbeauftragte/r Informatik		
Dozent(in):	Alle Professorinnen und Professoren der Fachgruppe		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	6	-	-
Lehrform / SWS:	Kolloquium zur Bachelorarbeit		
Gruppengröße:	Siehe § 22 der Bachelor-Rahmenprüfungsordnung		
Arbeitsaufwand:	90 Zeitstunden		
Leistungspunkte:	3		
Turnus:	Das Kolloquium zur Bachelorarbeit wird ca. 2 Wochen nach Abgabe der Bachelorarbeit durchgeführt.		
Teilnehmerzahl:	Wie Gruppengröße		
Anmeldungsmodalitäten:	Siehe § 19 PO und § 26 BRPO		
Voraussetzungen nach Prüfungsordnung:	Siehe § 26 BRPO		
Empfohlene Voraussetzungen (Modulprüfungen):	---		
Angestrebte Lernergebnisse:	Die/der Studierende ist in der Lage, die Ergebnisse der Bachelorarbeit, ihre fachlichen und methodischen Grundlagen, ihre fächerübergreifenden Zusammenhänge und ihre außerfachlichen Bezüge mündlich in begrenzter Zeit in einem Vortrag zu präsentieren. Darüber hinaus kann sie/er Fragen zu inhaltlichen Details, zu fachlichen Begründungen und Methoden sowie zu inhaltlichen Zusammenhängen zwischen Teilbereichen ihrer/seiner Arbeit selbstständig beantworten. Die/der Studierende kann ihre/seine Bachelorarbeit auch im Kontext beurteilen und ihre Bedeutung für die Praxis einschätzen und ist in der Lage, auch entsprechende Fragen nach themen- und fachübergreifenden Zusammenhängen zu beantworten.		
Inhalt:	Zunächst wird der Inhalt der Bachelorarbeit im Rahmen eines Vortrages präsentiert. Anschließend werden in einer Diskussion Fragen zum Vortrag und zur Bachelorarbeit gestellt, die von der/dem Studierenden beantwortet werden müssen.		

	Der Vortrag soll mindestens die Problemstellung der Bachelorarbeit, den gewählten Lösungsansatz, die erzielten Ergebnisse zusammen mit einer abschließenden Bewertung der Arbeit sowie einen Ausblick beinhalten. Je nach Thema können weitere Anforderungen hinzukommen, wie z.B. die vergleichende Darstellung alternativer oder konkurrierender Lösungsansätze, ein Literaturüberblick oder die Darlegung des aktuellen Standes der Wissenschaft. Die Dauer des Kolloquiums ist in § 26 der Bachelor-Rahmenprüfungsordnung und § 19 der Studiengangsprüfungsordnung geregelt.
Studien- / Prüfungsleistungen:	Siehe § 26 der Bachelor-Rahmenprüfungsordnung und § 19 der Studiengangsprüfungsordnung
Literatur:	• Kuzbari, Rafic; Ammer, Reinhard: Der wissenschaftliche Vortrag. Springer-Verlag Wien New York, 2006, 166 Seiten, ISBN: 978-3211235256 • Leopold-Wildburger, Ulrike: Verfassen und Vortragen - Wissenschaftliche Arbeiten und Vorträge leicht gemacht. 2. Auflage, Springer,2010. ISBN: 978-3642134197
Bemerkungen:	---

Kolloquium zur Bachelor-Arbeit Informatik

Kürzel:	KBIN
Untertitel:	Abschlussprüfung im Bachelor-Studium der Informatik
Studiensemester:	6. (Bachelor)
Modulverantwortliche(r):	Studiengangsbeauftragte/r Informatik
Dozent(in):	Alle Professoren der Fachgruppe Informatik
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 6; I/TI: 6; MI: -; WI: -
Lehrform / SWS:	Kolloquium zur Bachelor-Arbeit
Gruppengröße:	Siehe § 22 der Rahmenprüfungsordnung
Arbeitsaufwand:	90 Zeitstunden
Leistungspunkte:	3
Turnus:	Das Kolloquium zur Bachelor-Arbeit wird ca. 2 Wochen nach Abgabe der Bachelor-Arbeit durchgeführt.
Teilnehmerzahl:	Wie Gruppengröße
Anmeldungsmodalitäten:	Siehe § 26 der Rahmenprüfungsordnung und § 16 der Studiengangsprüfungsordnung
Voraussetzungen nach Prüfungsordnung:	Siehe § 26 der Rahmenprüfungsordnung und § 16 der Studiengangsprüfungsordnung
Empfohlene Voraussetzungen (Modulprüfungen):	---
Angestrebte Lernergebnisse:	Die/der Studierende ist in der Lage, die Ergebnisse der Bachelorarbeit, ihre fachlichen und methodischen Grundlagen, ihre fächerübergreifenden Zusammenhänge und ihre außerfachlichen Bezüge mündlich in begrenzter Zeit in einem Vortrag zu präsentieren. Darüber hinaus kann sie/er Fragen zu inhaltlichen Details, zu fachlichen Begründungen und Methoden sowie zu inhaltlichen Zusammenhängen zwischen Teilbereichen ihrer/seiner Arbeit selbstständig beantworten. Die/der Studierende kann ihre/seine Bachelor-Arbeit auch im Kontext beurteilen und ihre Bedeutung für die Praxis einschätzen und ist in der Lage, auch entsprechende Fragen nach themen- und fachübergreifenden Zusammenhängen zu beantworten.

Inhalt:	Zunächst wird der Inhalt der Bachelor-Arbeit im Rahmen eines Vortrages präsentiert. Anschließend werden in einer Diskussion Fragen zum Vortrag und zur Bachelor-Arbeit gestellt, die von der/dem Studierenden beantwortet werden müssen. Der Vortrag soll mindestens die Problemstellung der Bachelor-Arbeit, den gewählten Lösungsansatz, die erzielten Ergebnisse zusammen mit einer abschließenden Bewertung der Arbeit sowie einen Ausblick beinhalten. Je nach Thema können weitere Anforderungen hinzukommen, wie z.B. die vergleichende Darstellung alternativer oder konkurrierender Lösungsansätze, ein Literaturüberblick oder die Darlegung des aktuellen Standes der Wissenschaft. Die Dauer des Kolloquiums ist in § 26 der Rahmenprüfungsordnung und § 16 der Studiengangsprüfungsordnung geregelt.
Studien- / Prüfungsleistungen:	Siehe § 26 der Rahmenprüfungsordnung und § 16 der Studiengangsprüfungsordnung
Literatur:	Kuzbari, Rafic; Ammer, Reinhard: Der wissenschaftliche Vortrag. Springer-Verlag Wien New York, 2006, 166 Seiten, ISBN: 978-3211235256 Leopold-Wildburger, Ulrike: Verfassen und Vortragen - Wissenschaftliche Arbeiten und Vorträge leicht gemacht. 2. Auflage, Springer,2010. ISBN: 978-3642134197
Bemerkungen:	---

Logik und diskrete Strukturen

Kürzel:	LDS
Untertitel:	---
Studiensemester:	1. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Wolfram Conen
Dozent(in):	Prof. Dr. Wolfram Conen
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: 1 ID: 1 WI: 1
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Standard
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	---
Angestrebte Lernergebnisse:	Die Studierenden erkennen die grundlegende Bedeutung von diskreten Strukturen für Analyse, Darstellung und Lösung von Problemen in der Informatik. Sie beherrschen die elementaren automatisierten Beweisverfahren der Logik und können diese anwenden. Sie kennen die grundlegenden Begrifflichkeiten der Graphentheorie und können Probleme entsprechend darstellen. Ausgewählte Problemstellungen können sie lösen. Sie kennen und beherrschen die Grundzüge der RSA-Verschlüsselung (Zahlentheorie), von Entscheidungsbäume und bayes'schem Schliessen (Data Mining / Machine Learning).
Inhalt:	Historischer Abriss zur Entwicklung und Bedeutung der Logik für die Informatik (Frege, Russell, Hilbert, Gödel, Turing, Post) und zu den Grenzen der Berechenbarkeit. Exkurs: boole'sche Schaltkreise als Modell des Berechnens (inkl. Ausblick auf Funktionen und Logik).

	Grundlegende Begriffe und Konzepte der Mengenlehre (u.a. Eigenschaften von Funktionen, Abzählbarkeit) Logische Problemformulierung und Problemlösung (Aussagenlogik und Klassenkalkül 4/5, Datalog 1/5) Ausgewählte diskrete Strukturen und Probleme: Zahlentheorie (RSA), Entscheidungsbäume, diskrete Wahrscheinlichkeiten/Naive Bayes, Graphentheorie (Wegfindung), Kombinatorik (kombinatorische Explosion). Aufwand: Historie (10%), Mengen und Logik (60%), weitere diskrete Strukturen (30%)
Studien- / Prüfungsleistungen:	Klausur (75 Min.)
Literatur:	Skript, ergänzend: • Schöning: Logik für Informatiker, Spektrum • Schöning: Ideen der Informatik, Oldenbourg jeweils in aktueller Auflage.
Bemerkungen:	---

Logik und diskrete Strukturen

Kürzel:	LDS
Untertitel:	---
Studiensemester:	1. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Wolfram Conen
Dozent(in):	Prof. Dr. Wolfram Conen
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 1 I/TI: 1 MI: 1 WI: 1
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 58 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 122 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	---
Angestrebte Lernergebnisse:	Die Studierenden erkennen die grundlegende Bedeutung von diskreten Strukturen für Analyse, Darstellung und Lösung von Problemen in der Informatik. Sie beherrschen die elementaren automatisierten Beweisverfahren der Logik und können diese anwenden. Sie kennen die grundlegenden Begrifflichkeiten der Graphentheorie und können Probleme entsprechend darstellen. Ausgewählte Problemstellungen können sie lösen. Sie kennen und beherrschen die Grundzüge der RSA-Verschlüsselung (Zahlentheorie) und von Entscheidungsbäume (Data Mining / Machine Learning).

Inhalt:	• Historischer Abriss zur Entwicklung und Bedeutung der Logik für die Informatik (Frege, Russell, Hilbert, Gödel, Turing, Post). • Grundlegende Begriffe und Konzepte der Mengenlehre (u.a. Eigenschaften von Funktionen, Abzählbarkeit) • Logische Problemformulierung und Problemlösung (Aussagenlogik und Klassenkalkül 4/5, Datalog 1/5), • Ausgewählte diskrete Strukturen und Probleme: Zahlentheorie (RSA), Entscheidungsbäume, Graphentheorie (Wegfindung), Kombinatorik (kombinatorische Explosion; Abzählen). • Aufwand: Historie (10%), Mengen und Logik (60%), weitere diskrete Strukturen (30%)
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Klausur (75 Min.)
Literatur:	Skript, ergänzend: Schöning: Logik für Informatiker, Spektrum, Meinel: Mathematische Grundlagen der Informatik, Teubner, Schöning: Ideen der Informatik, Oldenbourg, jeweils in aktueller Auflage.
Bemerkungen:	---

Mathematik für Informatiker

Kürzel:	MIN
Untertitel:	---
Studiensemester:	2. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Laura Anderle
Dozent(in):	Prof. Dr. Laura Anderle
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 2 I/TI: 2 MI: - WI: -
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 54 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 126 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Grundlagen der Mathematik für Informatiker
Angestrebte Lernergebnisse:	Sicherer, praktischer Umgang mit den grundlegenden Begriffen des Integralkalküls, Sicherer Umgang mit den Begriffen der Linearen Algebra; Erlernen von strukturiertem Vorgehen durch Elemente der Algebra (Grundstrukturen)
Inhalt:	Integralrechnung, Flächenproblem, Begriff einer integrierbaren Funktion, Rechenregeln des bestimmten Integrals, Fundamentalsätze, Stammfunktionen, partielle Integration, Integration durch Substitution, Vektorbegriff, Skalarprodukt, Vektoren in Ebene und Raum, Projetionen, n-dimensionaler euklidischer Raum, Allgemeine Vektorräume, Unterraum, lineare Unabhängigkeit, Basis, Basistransformation, Norm, Metrik, metrischer Raum. Matrixbegriff, Matrixoperationen, Determinanten und Matrizen, Inverse Matrizen, lineare Gleichungssysteme, Matrixgleichungen, Homomorphismen, Bild und Kern,

	Eigenwerte, Eigenvektoren, Orthogonale Matrizen, Drehmatrizen
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Klausur (90 Min.)
Literatur:	Heuser: Lehrbuch der Analysis, Teil 1, Teubner Verlag, 2001 Timmann : Repetitorium der Analysis, Teil 1, Binomi Verlag, 2000 Anton: Lineare Algebra, Einführung, Grundlagen, Übungen, Spektrum Akademischer Verlag, 1998 Wille: Repetitorium der Linearen Algebra, Teil 1, Binomi-Verlag, 1998 Denecke: Algebra und Diskrete Mathematik für Informatiker, Teubner Verlag, 2003
Bemerkungen:	---

Mensch-Computer-Interaktion

Kürzel:	MCI
Untertitel:	
Studiensemester:	3. (Bachelor)
Modulverantwortliche(r):	N.N. MCI
Dozent(in):	N.N. MCI
Sprache:	Deutsch, Englisch bei Bedarf
Zuordnung zum Curriculum:	IN: 3 ID: 3 WI: 3
Lehrform / SWS:	3 SWS Vorlesung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt. Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 70 Zeitstunden Selbststudium: 110 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht Begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	EPR
Empfohlene Voraussetzungen (Modulprüfungen):	EPR, OPR

Angestrebte Lernergebnisse:	• Die Studierenden können einfache grafische Benutzeroberflächen mit einer aktuellen Programmiersprache für interaktive Anwendungen, z.B. JavaFX implementieren,
	  ○ indem sie Komponenten zur Ein- und Ausgabe, Ereignisbehandlung, Layouterzeugung, Benutzerführung und Eingabeprüfungen differenziert auswählen, programmatisch verstehen und zusammenführen,
	  ○ um später diese Aspekte sowohl in komplexere Anwendungen angepasst integrieren zu können als auch auf andere Programmiersprachen transferieren zu können.
	• Die Studierenden können einfache Benutzeroberflächen gestalten und kritisieren,
	  ○ indem Sie Grundlagen und Normen zur menschlichen Wahrnehmung und Kognition mit Grundsätzen und Normen der Interaktionsgestaltung, Usability, User Experience und weiteren Designprinzipien

	verbinden und mit Möglichkeiten der Ein- und Ausgabetechnologie zusammenführen, ○ um später konzeptionell bei der Gestaltung oder Evaluation von Benutzerobflächen Usability Probleme bewerten und letztlich vermeiden oder entdecken zu können. • Die Studierenden können die Phasen mensch-zentrierter Entwicklung auf definierte Problemstellungen anwenden ○ Indem Sie sie hierfür notwendige zentrale Methoden auswählen, diskutieren und differenzieren können, ○ Um später diese in eigens definierte Problemstellungen einführen und adaptieren zu können.
Inhalt:	• Grundlagen mensch-zentrierter Entwicklung sowie die hierfür zentralen verschiedenen Phasen und Methoden. • Theoretische Grundlagen: Sensorische Wahrnehmung, Mentale Modelle und Metaphern, Handlungsebenen und Modelle der Interaktion. • Interaktionsstile, Interaktionstechnologien und Interaktionsprinzipien. • Benutzerführung, Meldungen und Prüfung von Eingaben. • Barrierefreiheit • Grundlagen für die Programmierung von grafischen Benutzeroberflächen, insbesondere Ereignisbehandlung, Layout Komponenten, Interaktionselemente, Meldungen und Fehlerbehandlung.
Studien- / Prüfungsleistungen:	Klausur, mündliche Prüfung oder Kombinationsprüfung
Literatur:	• Heinecke A. M.: Mensch-Computer-Interaktion – Basiswissen für Entwickler und Gestalter. x.media.press, Springer, Berlin 2014. • Hartson, R., & Pyla, P. (2018). *The UX book: Agile UX design for a quality user experience*. Morgan Kaufmann. • Epple A.: JavaFX 8: Grundlagen und fortgeschrittene Techniken. dpunkt.verlag, Heidelberg 2015. • Offizielle Java Dokumentation (Oracle) sowie verschiedene, geprüfte und als Onlinematerialien hinterlegte Web-Tutorials zu JavaFX.
Bemerkungen:	—-

Mensch-Computer-Interaktion

Kürzel:	MCI
Untertitel:	---
Studiensemester:	3. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Andreas M. Heinecke
Dozent(in):	Prof. Dr. Andreas M. Heinecke, Prof. Dr. Jens Gerken, Dr.-Ing. Hansjürgen Paul
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 3 I/TI: 3 MI: - WI: -
Lehrform / SWS:	3 SWS Vorlesung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Module EPR, OPR, DSG
Angestrebte Lernergebnisse:	Studierende • verstehen die Bedeutung von Gebrauchstauglichkeit und Benutzererlebnis, • kennen die Grundbegriffe und die wesentlichen Modelle der Software-Ergonomie, • kennen und berücksichtigen die rechtlichen Anforderungen an Benutzungsschnittstellen, • kennen die relevanten Normen für die Informationsdarstellung und die Dialoggestaltung und können diese bei der Gestaltung von Benutzungsschnittstellen anwenden, • verstehen die software-technischen Grundlagen der GUI-Programmierung, insbesondere die Verwendung von Standard-

	Interaktionselementen und die Ereignisauswertung, • können Benutzungsoberflächen so konzipieren, dass sie die Anforderungen an die Gebrauchstauglichkeit erfüllen, • können formularbasierte Benutzungsschnittstellen für Java-Programme software-technisch angemessen implementieren.
Inhalt:	• Begriffe und Modelle der MCI • Software-Ergonomie und rechtliche Anforderungen • Handlungsprozesse und Fehler • Hardware für die Interaktion • Ein- / Ausgabe-Ebene • Dialog-Ebene • Werkzeug-Ebene • Benutzungsunterstützung • Interaktionselemente und Layout • Lexikalische, syntaktische, semantische und pragmatische Prüfung von Eingaben • MDI-Anwendungen, Menüs und Unterfenster • Undo und Redo • Formulare, Listen und Tabellen
Studien- / Prüfungsleistungen:	Studienleistungen als Voraussetzung zur Prüfungsteilnahme: keine (gemäß gesetzlicher Vorgabe) Prüfungsleistungen: Klausur am Rechner (120 Min.)
Literatur:	Heinecke A. M.: Mensch-Computer-Interaktion – Basiswissen für Entwickler und Gestalter. x.media.press, Springer, Berlin 2014. Herczeg M.: Software-Ergonomie - Theorien, Modelle und Kriterien für gebrauchstaugliche interaktive Computersysteme. Oldenbourg, München 2009. Epple A.: JavaFX 8: Grundlagen und fortgeschrittene Techniken. dpunkt.verlag, Heidelberg 2015. Steyer R.: Einführung in JavaFX: Moderne GUIs für RIAs und Java-Applikationen. Springer Vieweg, Wiesbaden 2014.
Bemerkungen:	---

Mathematische Grundlagen

Kürzel:	MGR		
Untertitel:			
Studiensemester:	1. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr. Laura Anderle		
Dozent(in):	Prof. Dr. Laura Anderle		
Sprache:	deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	1	1	1
Lehrform / SWS:	4 SWS Vorlesung, 1 SWS Übung		
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 40		
Arbeitsaufwand:	Kontaktzeit: 90 Zeitstunden Selbststudium: 90 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Wintersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Erscheinen zum ersten Vorlesungstermin, Anmeldung zu Übungsgruppen über den Moodlekurs zu diesem Modul		
Voraussetzungen nach Prüfungsordnung:	keine		
Empfohlene Voraussetzungen (Modulprüfungen):	keine		
Angestrebte Lernergebnisse:	Die Studierenden kennen und verstehen grundlegende Begriffe der Mathematik, insbesondere der Analysis, und deren Bedeutung in der Informatik. Sie können Rechentechniken von Hand und anhand einfacher Programmierung (Python) anwenden und einfache mathematische Modelle erstellen, interpretieren und anwenden.		
Inhalt:	• Zahlen: Zahlenräume der Mathematik und Zahlendarstellung im Rechner • Folgen: rekursiv und explizit definierte Folgen, vollständige Induktion, Grenzwertbestimmung, Konvergenzgeschwindigkeit • Funktionen: wichtige Modellfunktionen, Eigenschaften von Funktionen (Stetigkeit, Differenzierbarkeit, Krümmungsverhalten, Grenzwerte) und deren Bedeutung im Kontext von Modellbildung und Informatik), Taylorpolynome, Splines, exakte und numerische Integration		

	• Ausblick auf Anwendungen einfacher mathematischer Modelle in der Informatik
Studien- / Prüfungsleistungen:	Studienleistungen: Die Studierenden können während des Semesters Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der Vorlesung bekannt gegeben. Prüfungsleistungen: Klausur (90 Minuten)
Literatur:	• Weitz, E.: Konkrete Mathematik (nicht nur) für Informatiker • Papula, L.: Mathematik für Ingenieure und Naturwissenschaftler Band 1: Ein Lehr- und Arbeitsbuch für das Grundstudium
Bemerkungen:	Die Teilnahme am Mathematik-Vorkurs im Rahmen der Einstiegsakademie wird bei Bedarf empfohlen.

Objektorientierte Programmierung

Kürzel:	OPR
Untertitel:	---
Studiensemester:	2. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Marcel Luis
Dozent(in):	Prof. Dr. Marcel Luis
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 2 I/TI: 2 MI: 2 WI: 2
Lehrform / SWS:	3 SWS Vorlesung (davon 1 SWS seminaristisch), 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 143 Zeitstunden
Leistungspunkte:	7
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung
Angestrebte Lernergebnisse:	Die Studierenden kennen alle wesentlichen Konzepte der objektorientierten Programmierung. Sie kennen methodische Ansätze zur Entwicklung qualitativ guter, wartbarer und erweiterbarer Software und sind in der Lage, solche Lösungen mit den Mitteln der objektorientierten Programmierung zu erstellen.
Inhalt:	Klassenhierarchie und Polymorphie • Testautomatisierung • Collection-Klassen • Ausnahmen • Ein-/Ausgabe • Schnittstellen • Einführung in Entwurfsmuster • Reflection • Nebenläufigkeit • Lambda-Ausdrücke und Streams
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (120 Min.)

Literatur:	Guido Krüger, Heiko Hansen: Java-Programmierung -- Das Handbuch zu Java 8. O'Reilly, 2014. Christian Ullenboom: Java ist auch eine Insel. Rheinwerk Computing, 2014. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley: The Java Language Specification, Java SE 8 Edition. Oracle America, Inc., 2013. Martin Fowler: Refactoring, Improving the Design of Existing Code. Addison-Wesley, 2002.
Bemerkungen:	---

Objektorientierte Programmierung

Kürzel:	OPR
Untertitel:	Objektorientierte Konzepte, Lambdas und Streams verstehen und effektiv anwenden
Studiensemester:	2. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Marcel Luis
Dozent(in):	Prof. Dr. Marcel Luis
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: 2 ID: 2 WI: 2
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 40, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 70 Zeitstunden Selbststudium: 140 Zeitstunden
Leistungspunkte:	7
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung
Angestrebte Lernergebnisse:	• Die Studierenden kennen alle wesentlichen Konzepte der objektorientierten Programmierung sowie typische Problemstellungen, in denen diese sinnvoll und effektiv eingesetzt werden können. • Sie kennen darüber hinaus die aus der funktionalen Programmierung stammenden Konzepte der Lambdas und Streams, und sie wissen, wann diese vorteilhaft verwendet werden können. • Sie beherrschen den Umgang mit den gängigen Standardklassen (Collections, I/O) der Lehrsprache Java und verstehen die dahinter stehenden Konzepte. • Die Studierenden erkennen den Sinn und die Anwendung von Ausnahmen. • Sie erlernen das Schreiben von Unit-Tests als untrennbarem Bestandteil des Programmierablaufs. Sie verstehen, dass das Schreiben von Komponententests eine Form der Spezifikation des

	gewünschten Verhaltens ist und darum an den Anfang des Programmierablaufs gehört. • Insgesamt sind die Studierenden in der Lage, zu überschaubaren Aufgabenstellungen qualitativ gute, wartbare und erweiterbare Softwarelösungen zu erstellen.
Inhalt:	Klassenhierarchie und Polymorphie • Testautomatisierung mit JUnit • Collection-Klassen • Ausnahmen • Schnittstellen • Nutzen von Schnittstellen am Beispiel eines Entwurfsmusters • Lambda-Ausdrücke • Streams • Ein-/Ausgabe • Aufzählungstypen • Parallelität
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (120 Min.)
Literatur:	• Joachim Goll, Cornelia Heinisch: Java als erste Programmiersprache. Springer Vieweg, 2016. • Christian Ullenboom: Java ist auch eine Insel. Rheinwerk Computing, 2021. • Martin Fowler: Refactoring, Improving the Design of Existing Code. Addison Wesley, 2018. • Offizielle Spezifikation der jeweils aktuellen Java-Version als Nachschlagewerk
Bemerkungen:	---

Praxisphase

Kürzel:	PXP
Untertitel:	-
Studiensemester:	6. (Bachelor)
Modulverantwortliche(r):	Praxisphasen-Beauftragte/r des Fachbereichs Informatik
Dozent(in):	Alle Professoren der Fachgruppe Informatik
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 6 I/TI: 6 MI: 6 WI: 6
Lehrform / SWS:	Praktische Arbeit in einem Betrieb oder einer Einrichtung der Berufspraxis Erstellen eines Berichtes über die praktische Arbeit. Der Bericht soll auf folgende Punkte eingehen • Überblick über den Betrieb und das Arbeitsumfeld • Überblick über die Aktivitäten • Kritische Würdigung der Studieninhalte im Vergleich zu den Anforderungen im Betrieb Seminarvortrag über die Inhalte des Berichtes
Gruppengröße:	---
Arbeitsaufwand:	Die praktische Arbeit umfasst 12 Wochen (ca. 420 Zeitstunden) Erstellen Abschlussbericht und Vortrag: 30 Zeitstunden Ca. 450 Zeitstunden kreditierte Zeit.
Leistungspunkte:	15
Turnus:	Regulär: Sommersemester, jährlich Bei Bedarf und falls es organisatorisch möglich ist, Angebot auch im Wintersemester.
Teilnehmerzahl:	Nicht begrenzte Teilnehmerzahl
Anmeldungsmodalitäten:	Explizite Anmeldung im Prüfungsamt
Voraussetzungen nach Prüfungsordnung:	90 Leistungspunkte Alle Modulprüfungen der beiden ersten Semester müssen bestanden sein
Empfohlene Voraussetzungen (Modulprüfungen):	Es sollten alle Modulprüfungen des dritten Fachsemesters bestanden sein.

Angestrebte Lernergebnisse:	Die Praxisphase hat die Studierenden an die berufliche Tätigkeit des Informatikers durch konkrete Aufgabenstellung und praktische Mitarbeit in Betrieben oder anderen Einrichtungen der Berufspraxis herangeführt. Die Studierenden haben in Ansätzen gelernt, die im bisherigen Studium erworbenen Kenntnisse und Fähigkeiten anzuwenden und die bei der praktischen Tätigkeit gemachten Erfahrungen zu reflektieren und auszuwerten. Während der Praxisphase haben die Studierenden auch die verschiedenen Aspekte der betrieblichen Entscheidungsfindungsprozesse kennen gelernt und Einblick in informatische, technische, organisatorische, ökonomische und soziale Zusammenhänge des Betriebsgeschehens erhalten.
Inhalt:	Spezielle Inhalte für die Praxisphase werden nicht vorgegeben. Es muss lediglich sichergestellt sein, dass die Tätigkeit in der Praxisphase der Tätigkeit eines Informatikers, bzw. Medien- oder Wirtschaftsinformatikers entspricht. Um dies sicherzustellen, wird jeder Studierender vor und während der Praxisphase von einem Dozenten des Fachbereichs Informatik betreut. Dabei werden auch die geplanten Tätigkeiten besprochen.
Studien- / Prüfungsleistungen:	Bescheinigung des Arbeitgebers über die Tätigkeit im Unternehmen, keine Benotung Bescheinigung des Betreuers über das Erstellen und die Abgabe des Berichtes, keine Benotung
Literatur:	---
Bemerkungen:	---

Prozedurale Programmierung

Kürzel:	PPR
Untertitel:	---
Studiensemester:	3. bzw. 5. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Ulrike Griefahn
Dozent(in):	Lehrbeauftragte/r
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 5; I/TI: 3; MI: WP; WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium 70 Zeitstunden Eigenstudium 110 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul, Einteilung der Übungs- und Praktikumsgruppen erfolgt nach der 1. Vorlesungsstunde über den Moodle-Kurs
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Objektorientierte Programmierung, Algorithmen und Datenstrukturen
Angestrebte Lernergebnisse:	Die Studierenden kennen die Konzepte und Methoden der prozeduralen Programmierung und können diese effektiv und strukturiert bei der Entwicklung eigener prozeduraler Programme mit der Programmiersprache C einsetzen. Sie gehen sicher mit maschinennahen Konzepten wie Zeigern und Speicherverwaltung sowie mit Strukturen um. Die Studierenden sind damit in der Lage, sich zukünftig selbstständig und zügig in weitere prozedurale Sprachen einzuarbeiten.

Inhalt:	• Grundelemente von C • Funktionen und Speicherklassen • Präprozessor • Adressen und Zeiger • Dynamische Speicherverwaltung • Strukturen • Weitere ausgewählte Sprachelemente • Make • Überblick über die Erweiterungen von C zu C++
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.) Die Studierenden können während des Praktikums Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der 1. Vorlesungsstunde bekannt gegeben.
Literatur:	Manfred Dausmann, Ulrich Bröckl und Joachim Goll, C als erste Programmiersprache. Vom Einsteiger zum Profi. 8. überarb. und erw. Auflage, Springer Vieweg, 2014, 727 Seiten, ISBN-13: 978-3834818584 Jürgen Wolf, C von A bis Z. Galileo Computing, 3. Auflage, 2009, 1190 Seiten, ISBN-13: 978-3836214117 Vogt: C für Java-Programmierer, Carl Hanser Verlag 2007, 256 Seiten, ISBN-13: 978-3446407978
Bemerkungen:	---

Praxisphase

Kürzel:	PXP		
Untertitel:	–		
Studiensemester:	6. (Bachelor)		
Modulverantwortliche(r):	Praxisphasen-Beauftragte/r des Fachbereichs Informatik		
Dozent(in):	Alle Professorinnen und Professoren der Fachgruppe Informatik		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	6	6	6
Lehrform / SWS:	Praktische Arbeit in einem Betrieb oder einer Einrichtung der Berufspraxis Erstellen eines Berichtes über die praktische Arbeit. Der Bericht soll auf folgende Punkte eingehen • Überblick über den Betrieb und das Arbeitsumfeld • Überblick über die Aktivitäten • Kritische Würdigung der Studieninhalte im Vergleich zu den Anforderungen im Betrieb		
Gruppengröße:			
Arbeitsaufwand:	Die praktische Arbeit umfasst 12 Wochen (ca. 420 Zeitstunden) Erstellen des Abschlussberichts: 30 Zeitstunden Ca. 450 Zeitstunden kreditierte Zeit.		
Leistungspunkte:	15		
Turnus:	Regulär: Sommersemester, jährlich Bei Bedarf und falls es organisatorisch möglich ist, Angebot auch im Wintersemester.		
Teilnehmerzahl:	Nicht begrenzte Teilnehmerzahl		
Anmeldungsmodalitäten:	Explizite Anmeldung im Prüfungsamt		
Voraussetzungen nach Prüfungsordnung:	siehe § 15 PO und § 21 Bachelor-RahmenPO		
Empfohlene Voraussetzungen (Modulprüfungen):	Es sollten alle Modulprüfungen des dritten Fachsemesters bestanden sein.		
Angestrebte Lernergebnisse:	Die Praxisphase hat die Studierenden an die berufliche Tätigkeit des Informatikers bzw. an der Schnittstelle Wirtschaftsinformatik oder Informatik und Design durch konkrete Aufgabenstellung und praktische Mitarbeit in Betrieben oder anderen Einrichtungen der Berufspraxis herangeführt. Die Studierenden haben in Ansätzen gelernt, die im bisherigen Studium erworbenen		

	Kenntnisse und Fähigkeiten anzuwenden und die bei der praktischen Tätigkeit gemachten Erfahrungen zu reflektieren und auszuwerten. Während der Praxisphase haben die Studierenden auch die verschiedenen Aspekte der betrieblichen Entscheidungsfindungsprozesse kennen gelernt und Einblick in informatische, technische, organisatorische, ökonomische und soziale Zusammenhänge des Betriebsgeschehens erhalten.
Inhalt:	Spezielle Inhalte für die Praxisphase werden nicht vorgegeben. Es muss lediglich sichergestellt sein, dass die Tätigkeit in der Praxisphase der Tätigkeit eines Informatikers entspricht, bzw. eine Tätigkeit an der Schnittstelle Wirtschaftsinformatik oder Informatik und Design ist. Um dies sicherzustellen, wird jeder Studierende vor und während der Praxisphase von einem Professor oder einer Professorin des Fachbereichs Informatik betreut. Dabei werden auch die geplanten Tätigkeiten besprochen.
Studien- / Prüfungsleistungen:	siehe § 11 Bachelor-RahmenPO
Literatur:	—-
Bemerkungen:	—-

Rechnernetze

Kürzel:	REN
Untertitel:	Einführung in konvergente Netzwerke, insbesondere LANs und das Internet
Studiensemester:	2. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Detlef Mansel
Dozent(in):	Prof. Dr. Detlef Mansel
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 2 I/TI: 2 MI: - WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 54 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 96 Zeitstunden
Leistungspunkte:	5
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Keine, die Einteilung der Übungs- und Praktikumsgruppen wird in der 1. Vorlesungsstunde besprochen.
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Keine
Angestrebte Lernergebnisse:	Die Studierenden kennen den grundlegenden Aufbau konvergenter Netze. Sie kennen grundlegende Konzepte eines modernen LANs mit VLANs. Sie können beim Design, Aufbau und Betrieb eines mittelgroßen LANs unter Führung eines erfahrenen Netzadministrators eingesetzt werden. Darüber hinaus kennen Sie grundlegende Eigenschaften eines WAN und des Internet. Sie sind in der Lage, sich effektiv in weitere Aspekte von Netzwerken einschließlich Sicherheitsfragen und Management einzuarbeiten. Dazu sind Sie in der Lage, Protokolle höherer Schichten zügig zu erlernen und in das Schichtenmodell einzuordnen. Lehrsprache im Praktikum ist Cisco IOS.

Inhalt:	Grundbegriffe, Netztopologien , ISO/OSI-Schichtenmodell und Internet-Architektur • Übertragungsmedien und -kanäle, Bitübertragung und Codierung generisch und am Beispiel Ethernet • Schicht 2 Technologie am Beispiel Ethernet, LLC und MAC• Schicht 2 LAN Switching einschließlich VLANs und Spanning Tree • Internet-Adressierung sowie statisches und dynamisches Routing als Schicht 3 Technologie, Schicht 3 Routing im LAN • Grundlagen zu Weitverkehrsnetzen und zum Internet•
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistung einen Bonus im Einklang mit §11 der Rahmenprüfungsordnung: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss im Cisco-Labor gelöst und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (60 Min.)
Literatur:	Dye, McDonald, Rufi; Network Fundamentals, Cisco Press, 2007, ISBN 978-1-58713-208-7 Lewis; LAN Switching and Wireless, Cisco Press, 2008, ISBN 978-1- 58713-207-0 Graziani, Johnson; Routing Protocols and Concepts, Cisco Press, 2007, ISBN 978-1-58713-206-3 Vachon, Graziani; Accessing the WAN, Cisco Press, 2009, ISBN 978-1- 58713-205-6. Aktuelle Ergänzungen auf den Moodle-Kurs zu diesem Modul
Bemerkungen:	---

Rechnernetze

Kürzel:	REN
Untertitel:	
Studiensemester:	3. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Detlef Mansel
Dozent(in):	Prof. Dr. Detlef Mansel
Sprache:	deutsch
Zuordnung zum Curriculum:	IN: 3 ID: - WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: nicht begrenzt, Übung 40, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung für Übung und Praktikum via Moodle
Voraussetzungen nach Prüfungsordnung:	keine
Empfohlene Voraussetzungen (Modulprüfungen):	keine
Angestrebte Lernergebnisse:	• Die Studierenden kennen den grundlegenden Aufbau konvergenter Netze. Sie kennen grundlegende Konzepte eines modernen LANs mit VLANs. • Sie können beim Design, Aufbau und Betrieb eines mittelgroßen LANs unter Führung eines erfahrenen Netzadministrators eingesetzt werden. • Darüber hinaus kennen Sie grundlegende Eigenschaften eines WAN und des Internets. • Sie sind in der Lage, sich effektiv in weitere Aspekte von Netzwerken einschließlich Sicherheitsfragen und Management einzuarbeiten. Darüber hinaus sind Sie in der Lage, Protokolle höherer Schichten zügig zu erlernen und in das Schichtenmodell einzuordnen. • Lehrsprache im Praktikum ist Cisco IOS.
Inhalt:	Grundbegriffe, Netztopologien , ISO/OSI-Schichtenmodell und Internet-Architektur • Übertragungsmedien und -kanäle, Bitübertragung und Codierung generisch und am Beispiel Ethernet

	• Schicht 2 Technologie am Beispiel Ethernet, LLC und MAC• Schicht 2 LAN Switching einschließlich VLANs und Spanning Tree • Internet-Adressierung sowie statisches und dynamisches Routing als Schicht 3 Technologie, Schicht 3 Routing im LAN • Grundlagen zu Weitverkehrsnetzen und zum Internet • Einführung zu TCP und UDP und well-known-Port Anwendungsschichtprotokollen
Studien- / Prüfungsleistungen:	Studienleistungen: Die Studierenden können während des Praktikums Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der Vorlesung bekannt gegeben. Prüfungsleistungen: Klausur
Literatur:	• Riggert/ Lübben, Rechnernetze, Hanser Verlag, aktuellste Auflage (online-Ressource) • Dye, McDonald, Rufi; Network Fundamentals, Cisco Press, 2007, ISBN 978-1-58713-208-7 • Lewis; LAN Switching and Wireless, Cisco Press, 2008, ISBN 978-1- 58713-207-0 • Graziani, Johnson; Routing Protocols and Concepts, Cisco Press, 2007, ISBN 978-1-58713-206-3 • Vachon, Graziani; Accessing the WAN, Cisco Press, 2009, ISBN 978-1- 58713-205-6 • Aktuelle Ergänzungen auf den Moodle-Kurs zu diesem Modul
Bemerkungen:	-

Robotik

Kürzel:	ROB
Untertitel:	Grundlegende Konzepte und Verfahren zur Entwicklung von Applikationen für Roboter / Manipulatoren
Studiensemester:	4. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Hartmut Surmann
Dozent(in):	Prof. Dr. Hartmut Surmann
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: 4; MI: WP; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	keine
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung, Grundlagen der Mathematik für Informatiker, Lineare Algebra
Angestrebte Lernergebnisse:	Die Studierenden kennen die Begriffe und Komponenten von Industrierobotern sowie die Konzepte und Methoden der Programmierung und können diese effektiv und strukturiert bei der Entwicklung eigener Steuerungsprogramme einsetzen. Sie gehen sicher mit maschinennahen Steuerungen und Regelungen um und wissen, welchen Einfluss die Steuerung und Regelung auf das Roboterverhalten hat. Sie kennen die Gefahren und Wichtigkeit der Einhaltung von Vorschriften sowohl auf technischer als auch sozialer Ebene beim Umgang mit Manipulatoren. Die Studierenden sind zudem in der Lage, sich selbstständig und zügig in unterschiedliche Industrieroboter und deren Programmierumgebung

	einzuarbeiten. Lehrsprachen sind KUKA Robot Programming Language, C / C++, Python.
Inhalt:	Grundlagen der Industrierobotik / Manipulatortechnik •Begriffsbildung Komponenten eines Manipulators / Industrieroboters • Beschreibung einer Roboterstellung • Transformation zwischen Roboter- und Weltkoordinaten • Bewegungsart und Interpolation • Roboterprogrammierung
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• Wolfgang Weber: Industrieroboter, Methoden der Steuerung und Regelung, Hanser Verlag, 2. Auflage, ISBN 978-3-446-41031-2 • Stefan Hesse: Grundlagen der Handhabungstechnik, Hanser Verlag, ISBN 978-3-446-40473-1 • Bruno Siciliano, Oussama Khatib (Eds.): Handbook of Robotic, ISBN 978-3-540-23957-4 • Craig, J.J. (2004), "Introduction to Robotics: Mechanics and Control (3rd Edition)", 8, 2004. Prentice Hall
Bemerkungen:	---

Statistik und Lineare Algebra

Kürzel:	SLA		
Untertitel:			
Studiensemester:	2. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr. Laura Anderle		
Dozent(in):	Prof. Dr. Laura Anderle		
Sprache:	deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	2	2	2
Lehrform / SWS:	4 SWS Vorlesung, 1 SWS Übung		
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 40		
Arbeitsaufwand:	Kontaktzeit: 75 Zeitstunden Selbststudium: 105 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Sommersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Erscheinen zum ersten Vorlesungstermin, Anmeldung zu Übungsgruppen über den Moodlekurs zu diesem Modul		
Voraussetzungen nach Prüfungsordnung:	keine		
Empfohlene Voraussetzungen (Modulprüfungen):	keine		
Angestrebte Lernergebnisse:	Sicherer, praktischer Umgang mit den Grundbegriffen der Wahrscheinlichkeitsrechnung und Statistik; Sicherer Umgang mit den Begriffen der Linearen Algebra; Erlernen von strukturiertem Vorgehen durch Elemente der Algebra (Grundstrukturen).		
Inhalt:	• Wahrscheinlichkeitsbegriff, Elemente der Kombinatorik, bedingte Wahrscheinlichkeit, Verteilung • Verteilungsfunktionen und Dichten, Parameter von Verteilungen, Eigenschaften der Normalverteilung • Grundelemente der beschreibenden Statistik • Vektorbegriff, Skalarprodukt, Vektoren in Ebene und Raum • Lineare Gleichungssysteme, Gauß-Algorithmus • Lineare Abbildungen und Geometrie, Matrizen • Skalar- und Kreuzprodukt, Abstandsmaße, Determinanten		

	• Definitheit von Matrizen, spezielle lineare Abbildungen, Eigenwerte und Eigenvektoren
Studien- / Prüfungsleistungen:	Studienleistungen: Die Studierenden können während des Semesters Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der Vorlesung bekannt gegeben. Prüfungsleistungen: Klausur (90 Minuten)
Literatur:	• Sysdæter, Hammond: Mathematik für Wirtschaftswissenschaftler, Basiswissen mit Praxisbezug, Pearson-Verlag, 2006. • Greiner, Tinhofer: Stochastik für Studienanfänger der Informatik, Hanser Verlag, 1996. • Irle: Wahrscheinlichkeitstheorie und Statistik, Grundlagen – Resultate – Anwendungen, Teubner-Verlag, 2001 • Nollau et.al.: Wahrscheinlichkeitsrechnung und Statistik in Beispielen und Aufgaben, Teubner-Verlag, 2002) • Heuser: Lehrbuch der Analysis, Teil 1, Teubner Verlag, 2001 • Timmann : Repetitorium der Analysis, Teil 1, Binomi Verlag, 2000 • Anton: Lineare Algebra, Einführung, Grundlagen, Übungen, Spektrum Akademischer Verlag, 1998 • Wille: Repetitorium der Linearen Algebra, Teil 1, Binomi-Verlag, 1998 • Denecke: Algebra und Diskrete Mathematik für Informatiker, Teubner Verlag, 2003
Bemerkungen:	---

Softwareprojekt Informatik

Kürzel:	SPIN
Untertitel:	---
Studiensemester:	4. und 5. (Bachelor)
Modulverantwortliche(r):	Studiengangsbeauftragte/r Informatik
Dozent(in):	Alle Professoren der Fachgruppe Informatik
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 4/5; I/TI: 4/5; MI: -; WI: -
Lehrform / SWS:	4. Sem.: 2 SWS Vorlesung, 2 SWS Übung, 1 SWS Praktikum; 5. Sem.: 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: Projektteams von 5 bis 8 Studierenden
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 70 Zeitstunden Eigenstudium: 290 Zeitstunden
Leistungspunkte:	12
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Explizite Voranmeldung und Anmeldung erforderlich. Informationen im Info-Center Informatik unter Bachelorprojekt.
Voraussetzungen nach Prüfungsordnung:	Einführung in die Programmierung, Objektorientierte Programmierung, Algorithmen und Datenstrukturen, unbenotetes Testat oder unbenoteter Vortrag, regelmäßige Anwesenheit bei Projektbesprechungen
Empfohlene Voraussetzungen (Modulprüfungen):	Die Inhalte der Module Mensch-Computer-Interaktion, und Softwaretechnik gehören zusätzlich für jedes Projekt zu den Minimalvoraussetzungen. Die Inhalte der Module Datenbanksysteme, Internetsprachen und/oder Prozedurale Programmierung können abhängig von der Studienrichtung für die meisten weiteren Projekte Voraussetzung sein. Projektspezifisch kann zudem jedes Modul bis inklusive des 3. Semesters Voraussetzung für ein Projektthema sein.
Angestrebte Lernergebnisse:	Die Studierenden sind in der Lage, durch wissenschaftliches Vorgehen für praktische Problemstellungen den Stand der Technik zu

	recherchieren, Anforderungen zu analysieren, Lösungen zu entwickeln und zu begründen sowie Arbeitsergebnisse professionell zu präsentieren und zu bewerten. Sie können die in der Vorlesung zu diesem Modul erlernten grundlegenden Management-Methoden zur Projektdefinition, -planung und -kontrolle bei der Projektarbeit anwenden und sind in der Lage, Besprechungen zu moderieren und zu protokollieren. Die Studierenden haben ein Grundverständnis für die Aufgaben und Erfolgsfaktoren bei der Durchführung eines mittelgroßen Software-Projekts in einem Team. Sie sind in der Lage das bisher im Studium Erlernte – insbesondere Methoden, Verfahren und Werkzeuge – anzuwenden, um ein komplexes Softwareprojekt von der Anforderungsanalyse über Entwurf, Implementierung und Evaluierung bis hin zur Auslieferung selbstständig und im Team von 5 bis 8 Studierenden zu bewältigen. Die Studierenden können komplexe Aufgaben sinnvoll strukturieren und typische Schnittstellenprobleme sowohl auf technisch-fachlicher als auch auf sozialer Ebene bewältigen.
Inhalt:	Der Vorlesungsteil wird als globale Veranstaltung für alle Teilnehmer abgehalten und führt in die Grundlagen des wissenschaftlichen Arbeitens und des Managements von Softwareprojekten ein. Zum wissenschaftlichen Arbeiten gehören: • Recherche • Analyse • Erstellen wissenschaftlicher Texte • Präsentation Der Vorlesungsteil wird als globale Veranstaltung für alle Teilnehmer abgehalten und führt in die Grundlagen des Managements von Softwareprojekten ein. Hierzu gehören: • Dateiorganisation, Protokolle • Projektdefinition • Projektplanung • Konfigurationsmanagement • Projektkontrolle und -steuerung • Projektabschluss Im Praktikumsteil steht die systematische Anwendung und Zusammenführung von in

	Vorgängerveranstaltungen erlerntem Wissen im Vordergrund: • Durchführung eines mittelgroßen und anspruchsvollen Software-Projekts • Selbstständige Durchführung des Projekts von der Analyse über Design, Implementierung und Test bis zur Dokumentation • Anwendung von grundlegenden Projektmanagement-Methoden für Definition, Planung, Kontrolle und Realisierung des Projekts. • Vertiefung von Programmierkenntnissen • Nutzung von Versionsmanagementwerkzeugen und Ticketsystemen • Softwareentwicklung im Team und ggf. unter Beteiligung von externen Anwendern • In regelmäßigen Projektsitzungen werden im Rahmen einer Qualitätssicherung die Zwischenergebnisse von den Teams durch Präsentation und Vorführung vorgestellt und diskutiert. Die Projektthemen werden rechtzeitig vor Beginn der Veranstaltung bekannt gemacht. Es wird versucht, praxisnahe Projekte, ggf. auch von hochschulexternen Anwendern der praktischen und technischen Informatik zu akquirieren. Projektvorschläge von Studierenden sind nach Absprache ebenfalls möglich.
Studien- / Prüfungsleistungen:	Unbenotetes Testat oder Vortrag zu den Inhalten des Vorlesungsteils als Voraussetzung zur Prüfungszulassung Prüfungsleistungen: Ausarbeitung in Form einer entwickelten Software, Ausarbeitungen und Präsentationen der geforderten Projektergebnisse.
Literatur:	Theisen, Manuel René, Wissenschaftliches Arbeiten: Erfolgreich bei Bachelor- und Masterarbeit, 17. aktualis. und bearb. Aufl., 2017, Verlag Franz Vahlen GmbH, 320 Seiten, ISBN: 978-3-8006-5382-9 Burghardt, Manfred, Einführung in Projektmanagement: Definition, Planung, Kontrolle und Abschluss, 6. aktualis. und erw. Aufl., 2013, Publicis Corporate Publishing, 391 Seiten, ISBN: 978-3895784002 Helmut Balzert, Lehrbuch der Software-Technik – Software- Management, Software-Qualitätssicherung, Unternehmensmodellierung, Band 2, 2. Auflage, Spektrum Akademischer Verlag, 2008, 721 Seiten, ISBN: 978-3827411617

Bemerkungen:	Das Software-Projekt wird über zwei Semester durchgeführt. Ein Großteil der Bearbeitung soll in Absprache mit der Projektgruppe während der vorlesungsfreien Zeit nach dem Sommersemester erfolgen, so dass das Projektende bereits zu Beginn des folgenden Wintersemesters erreicht werden kann.

Softwaretechnik

Kürzel:	SWT
Untertitel:	Requirements Engineering und Objektorientierte Analyse
Studiensemester:	3. (Bachelor)
Modulverantwortliche(r):	Prof. Jürgen Znotka
Dozent(in):	Prof. Jürgen Znotka
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 3 I/TI: 3 MI: 3 WI: 3
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 53 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 97 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Grundlagen der Mathematik für Informatiker,
Angestrebte Lernergebnisse:	Die Studierenden kennen: • Begriffe der Softwaretechnik wie Prozessmodell, Analyse und Design, Softwarewartung, Testen, Lastenheft, Pflichtenheft, SRS (Software Requirements Specification) und Softwarequalität • Begriffe der Objektorientierung wie Klasse, Objekt, Attribut, Operation, Assoziation, Aggregation, Komposition, Generalisierung / Spezialisierung und • die folgenden Diagramme der UML: Klassendiagramm, Anwendungsfalldiagramm,

	Aktivitätsdiagramm, Sequenzdiagramm, Objektdiagramm und Zustandsdiagramm • Begriffe der Softwarequalität wie Functionality, Usability, Reliability, Portability und Supportability (FURPS) Die Studierenden verstehen: • den Zusammenhang der einzelnen Phasen in verschiedenen Softwareprozessen und die jeweiligen Vor- und Nachteile • den Zusammenhang zwischen Anforderungen und objektorientierten Modellen • Die Studierenden können das Erlernte anwenden, um • aus unstrukturierten Anforderungen an ein System funktionale Anforderungen zu extrahieren • qualitative Anforderungen zu formulieren • objektorientierte Modelle auf Basis der UML zu erstellen für verschiedene Anwendungsdomänen
Inhalt:	• Einführung in die Softwaretechnik (1) • Software Prozesse (1) • Requirements Engineering (3) • Systemmodellierung (1) • Objektorientierte Systemanalyse (2) • Softwarewartung (1) • Softwaretest (1) • UML Diagramme (4)
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (60 Min.)
Literatur:	Sommerville, Ian: Software Engineering, Addison-Wesley, 10th Edition, 2015 Oestereich, Bernd: Analyse und Design mit der UML, 10. Auflage, 2012, Oldenbourg Verlag
Bemerkungen:	---

Softwareprojekt Informatik

Kürzel:	SPIN
Untertitel:	---
Studiensemester:	4. und 5. (Bachelor)
Modulverantwortliche(r):	Studiengangsbeauftragte/r Informatik
Dozent(in):	Alle Professorinnen und Professoren der Fachgruppe Informatik
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: 4+5 ID: - WI: -
Lehrform / SWS:	4. Sem.: 2 SWS Vorlesung, 2 SWS Übung, 1 SWS Praktikum; 5. Sem.: 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 40, Praktikum: Projektteams von 5 bis 8 Studierenden
Arbeitsaufwand:	Kontaktzeit: 85 Zeitstunden Selbststudium: 275 Zeitstunden
Leistungspunkte:	12
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Explizite Voranmeldung und Anmeldung erforderlich. Informationen im Info-Center Informatik unter Bachelorprojekt.
Voraussetzungen nach Prüfungsordnung:	Einführung in die Programmierung, Objektorientierte Programmierung, Algorithmen und Datenstrukturen
Empfohlene Voraussetzungen (Modulprüfungen):	Die Inhalte der Module Mensch-Computer-Interaktion, und Softwaretechnik gehören zusätzlich für jedes Projekt zu den Minimalvoraussetzungen. Die Inhalte der Module Datenbanksysteme, Internetsprachen und/oder Prozedurale Programmierung können abhängig von der Studienrichtung für die meisten weiteren Projekte Voraussetzung sein. Projektspezifisch kann zudem jedes Modul bis inklusive des 3. Semesters Voraussetzung für ein Projektthema sein.
Angestrebte Lernergebnisse:	Die Studierenden sind in der Lage, durch wissenschaftliches Vorgehen für praktische Problemstellungen den Stand der Technik zu recherchieren, Anforderungen zu analysieren, Lösungen zu entwickeln und zu begründen sowie Arbeitsergebnisse professionell zu präsentieren und zu bewerten.

	Sie können die in der Vorlesung zu diesem Modul erlernten grundlegenden Management-Methoden zur Projektdefinition, -planung und -kontrolle bei der Projektarbeit anwenden und sind in der Lage, Besprechungen zu moderieren und zu protokollieren. Die Studierenden haben ein Grundverständnis für die Aufgaben und Erfolgsfaktoren bei der Durchführung eines mittelgroßen Software-Projekts in einem Team. Sie sind in der Lage das bisher im Studium Erlernte – insbesondere Methoden, Verfahren und Werkzeuge – anzuwenden, um ein komplexes Softwareprojekt von der Anforderungsanalyse über Entwurf, Implementierung und Evaluierung bis hin zur Auslieferung selbstständig und im Team von 5 bis 8 Studierenden zu bewältigen. Die Studierenden können komplexe Aufgaben sinnvoll strukturieren und typische Schnittstellenprobleme sowohl auf technisch-fachlicher als auch auf sozialer Ebene bewältigen.
Inhalt:	Der Vorlesungsteil wird als globale Veranstaltung für alle Teilnehmer abgehalten und führt in die Grundlagen des wissenschaftlichen Arbeitens und des Managements von Softwareprojekten ein. Zum wissenschaftlichen Arbeiten gehören: • Recherche • Analyse • Erstellen wissenschaftlicher Texte • Präsentation Der Vorlesungsteil wird als globale Veranstaltung für alle Teilnehmer abgehalten und führt in die Grundlagen des Managements von Softwareprojekten ein. Hierzu gehören: • Dateiorganisation, Protokolle • Projektdefinition • Projektplanung • Konfigurationsmanagement • Projektkontrolle und -steuerung • Projektabschluss Im Praktikumsteil steht die systematische Anwendung und Zusammenführung von in Vorgängerveranstaltungen erlerntem Wissen im Vordergrund:

	• Durchführung eines mittelgroßen und anspruchsvollen Software-Projekts • Selbstständige Durchführung des Projekts von der Analyse über Design, Implementierung und Test bis zur Dokumentation • Anwendung von grundlegenden Projektmanagement-Methoden für Definition, Planung, Kontrolle und Realisierung des Projekts. • Vertiefung von Programmierkenntnissen • Nutzung von Versionsmanagementwerkzeugen und Ticketsystemen • Softwareentwicklung im Team und ggf. unter Beteiligung von externen Anwendern • In regelmäßigen Projektsitzungen werden im Rahmen einer Qualitätssicherung die Zwischenergebnisse von den Teams durch Präsentation und Vorführung vorgestellt und diskutiert. Die Projektthemen werden rechtzeitig vor Beginn der Veranstaltung bekannt gemacht. Es wird versucht, praxisnahe Projekte, ggf. auch von hochschulexternen Anwendern der praktischen und technischen Informatik zu akquirieren. Projektvorschläge von Studierenden sind nach Absprache ebenfalls möglich.
Studien- / Prüfungsleistungen:	Unbenotetes Testat oder Vortrag zu den Inhalten des Vorlesungsteils als Voraussetzung zur Prüfungszulassung Prüfungsleistungen: Ausarbeitung in Form einer entwickelten Software, Ausarbeitungen und Präsentationen der geforderten Projektergebnisse.
Literatur:	• Theisen, Manuel René, Wissenschaftliches Arbeiten: Erfolgreich bei Bachelor- und Masterarbeit, 17. aktualis. und bearb. Aufl., 2017, Verlag Franz Vahlen GmbH, 320 Seiten, ISBN: 978-3-8006-5382-9 • Burghardt, Manfred, Einführung in Projektmanagement: Definition, Planung, Kontrolle und Abschluss, 6. aktualis. und erw. Aufl., 2013, Publicis Corporate Publishing, 391 Seiten, ISBN: 978-3895784002 • Helmut Balzert, Lehrbuch der Software-Technik – Software- Management, Software-Qualitätssicherung, Unternehmensmodellierung, Band 2, 2. Auflage, Spektrum Akademischer Verlag, 2008, 721 Seiten, ISBN: 978-3827411617
Bemerkungen:	Das Software-Projekt wird über zwei Semester durchgeführt. Ein Großteil der Bearbeitung soll in Absprache mit der Projektgruppe während der vorlesungsfreien Zeit nach dem Sommersemester

	erfolgen, so dass das Projektende bereits zu Beginn des folgenden Wintersemesters erreicht werden kann.

Softwaretechnik

Kürzel:	SWT		
Untertitel:	Requirements Engineering und Objektorientierte Analyse		
Studiensemester:	3. (Bachelor)		
Modulverantwortliche(r):	Prof. Jürgen Znotka		
Dozent(in):	Prof. Jürgen Znotka		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	3	3	3
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 Praktikum		
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20		
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Wintersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Grundlagen der Mathematik für Informatiker		
Angestrebte Lernergebnisse:	Die Studierenden kennen: • Begriffe der Softwaretechnik wie Prozessmodell, Analyse und Design, Softwarewartung, Testen, Lastenheft, Pflichtenheft, SRS (Software Requirements Specification) und Softwarequalität • Begriffe der Objektorientierung wie Klasse, Objekt, Attribut, Operation, Assoziation, Aggregation, Komposition, Generalisierung / Spezialisierung und • die folgenden Diagramme der UML: Klassendiagramm, Anwendungsfalldiagramm,		

	Aktivitätsdiagramm, Sequenzdiagramm, Objektdiagramm und Zustandsdiagramm • Begriffe der Softwarequalität wie Functionality, Usability, Reliability, Portability und Supportability (FURPS) Die Studierenden verstehen: • den Zusammenhang der einzelnen Phasen in verschiedenen Softwareprozessen und die jeweiligen Vor- und Nachteile • den Zusammenhang zwischen Anforderungen und objektorientierten Modellen • Die Studierenden können das Erlernte anwenden, um • aus unstrukturierten Anforderungen an ein System funktionale Anforderungen zu extrahieren • qualitative Anforderungen zu formulieren • objektorientierte Modelle auf Basis der UML zu erstellen für verschiedene Anwendungsdomänen
Inhalt:	• Einführung in die Softwaretechnik (1) • Software Prozesse (1) • Requirements Engineering (3) • Systemmodellierung (1) • Objektorientierte Systemanalyse (2) • Softwarewartung (1) • Softwaretest (1) • UML Diagramme (4)
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (60 Min.)
Literatur:	• Sommerville, Ian: Software Engineering, Addison-Wesley, 10th Edition, 2015 • Oestereich, Bernd: Analyse und Design mit der UML, 10. Auflage, 2012, Oldenbourg Verlag
Bemerkungen:	---

Technische Grundlagen der Informatik

Kürzel:	TGI
Untertitel:	Grundlegende Konzepte zum Aufbau und der Funktionsweise der Hardware von Rechnern
Studiensemester:	1. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Detlef Mansel
Dozent(in):	Prof. Dr. Detlef Mansel
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 1 I/TI: 1 MI: 1 WI: -
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 54 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 96 Zeitstunden
Leistungspunkte:	5
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über das Moodle-System zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	keine
Angestrebte Lernergebnisse:	Die Studierenden kennen den Aufbau und die Funktionsweise der Hardware von Rechnern
Inhalt:	• Historische Einführung • Zahlendarstellung, Konvertierungs- und arithmetische Algorithmen • Boolsche Schaltalgebra • Technische Realisierung von Schaltfunktionen, Transistoren und Gatter • Kombinatorische Schaltkreise, Schaltnetze • Speicherelemente, Latches, Flip-Flops, sequentielle Schaltungen • Endliche Zustandsautomaten und Steuerwerke • Operationswerke • Komplexe Schaltwerke, Aufgabenspezifische Mikroprozessoren, Universelle

	Mikroprozessoren, Einführung eines Modellprozessors • Speicher und deren Realisierung • Rechnerarchitekturen • Computerperipherie
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (60 Min.)
Literatur:	Dirk. W. Hoffmann: Grundlagen der Technischen Informatik ; Hanser-Verlag, 2014, ISBN 978-3-446-44251-1 Tobias Häberlein: Technische Informatik : Ein Tutorium der Maschinenprogrammierung und Rechnertechnik ; Vieweg + Teubner, 2011, ISBN 978-3-8348-1372-5
Bemerkungen:	---

Technisches Englisch für Informatiker

Kürzel:	TENI
Untertitel:	---
Studiensemester:	1. (Bachelor)
Modulverantwortliche(r):	Dr. Petra Iking – Leiterin des Sprachenzentrums
Dozent(in):	Bernd Winkelräth
Sprache:	Englisch
Zuordnung zum Curriculum:	I/PI: 1; I/TI: 1; MI: -; WI: -
Lehrform / SWS:	Seminaristische Veranstaltung im Präsenzstudium und angeleitetes Selbststudium (ggf. im MultiMedia Sprachlabor des Sprachenzentrums) / 4 SWS
Gruppengröße:	≤ 30
Arbeitsaufwand:	150 Std. Präsenz- und Selbststudium
Leistungspunkte:	5
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	≤ 30
Anmeldungsmodalitäten:	Online unter www.spz.w-hs.de im Klausurzeitraum, der dem jeweiligen Semester vorausgeht. Genaue Daten sind den Aushängen und der Homepage des SPZ zu entnehmen.
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Fortgeschrittene Englischkenntnisse auf dem Niveau der Jahrgangsstufe 11/12 ; ggf. zusätzlich erfolgreich abgeschlossener Auffrischungskurs Englisch bzw. Teilnahme am „English Support Programme (ESP)“ des Sprachenzentrums
Angestrebte Lernergebnisse:	Die Studierenden erwerben berufsorientierte englischsprachige Diskurs- und Handlungskompetenz unter Berücksichtigung (inter-)kultureller Elemente.
Inhalt:	Die Veranstaltung führt in die Fachsprache anhand ausgewählter Inhalte z.B. aus folgenden Bereichen ein : AI (Artificial Intelligence), Basic Geometric and Mathematical Terminology, Biometric Systems, Diagrammatic Representation, Display Technology, Networking, Online Security Threats, Robotics, SDLC (Software Development Life Cycle).
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Klausur (120 Min.)

Literatur:	Wird in der ersten Veranstaltung bekannt gegeben
Bemerkungen:	Selbststudienelemente im MulitMedia-Sprachlabor des Sprachenzentrums

Technisches Englisch für Informatiker

Kürzel:	TENI		
Untertitel:			
Studiensemester:	1 (Bachelor)		
Modulverantwortliche(r):	Dr. Petra Iking – Leiterin des Sprachenzentrums		
Dozent(in):	Bernd Winkelräth		
Sprache:	Englisch		
Zuordnung zum Curriculum:	IN	ID	WI
	1	-	-
Lehrform / SWS:	Seminaristische Veranstaltung im Präsenzstudium und angeleitetes Selbststudium (ggf. im MultiMedia Sprachlabor des Sprachenzentrums) / 4 SWS		
Gruppengröße:	≤ 30		
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 90 Zeitstunden		
Leistungspunkte:	5		
Turnus:	Wintersemester, jährlich		
Teilnehmerzahl:	≤ 30		
Anmeldungsmodalitäten:	Online unter www.spz.w-hs.de im Klausurzeitraum, der dem jeweiligen Semester vorausgeht. Genaue Daten sind den Aushängen und der Homepage des SPZ zu entnehmen.		
Voraussetzungen nach Prüfungsordnung:	keine		
Empfohlene Voraussetzungen (Modulprüfungen):	Fortgeschrittene Englischkenntnisse auf dem Niveau der Jahrgangsstufe 11/12 ; ggf. zusätzlich erfolgreich abgeschlossener Auffrischungskurs Englisch bzw. Teilnahme am „English Support Programme (ESP)“ des Sprachenzentrums		
Angestrebte Lernergebnisse:	Die Studierenden erwerben berufsorientierte englischsprachige Diskurs- und Handlungskompetenz unter Berücksichtigung (inter-)kultureller Elemente.		
Inhalt:	Die Veranstaltung führt in die Fachsprache anhand ausgewählter Inhalte z.B. aus folgenden Bereichen ein : AI (Artificial Intelligence), Basic Geometric and Mathematical Terminology, Biometric Systems, Diagrammatic Representation, Display Technology, Networking, Online Security Threats, Robotics, SDLC (Software Development Life Cycle).		
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Klausur (120 Min.)		
Literatur:	Wird in der ersten Veranstaltung bekannt gegeben		

Bemerkungen:	Selbststudienelemente im MulitMedia-Sprachlabor des Sprachenzentrums

Technische Grundlagen der Informatik

Kürzel:	TGI
Untertitel:	
Studiensemester:	1. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Detlef Mansel
Dozent(in):	Prof. Dr. Detlef Mansel
Sprache:	deutsch
Zuordnung zum Curriculum:	IN: 1 ID: - WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: nicht begrenzt, Übung 40, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung für Übung und Praktikum via Moodle
Voraussetzungen nach Prüfungsordnung:	keine
Empfohlene Voraussetzungen (Modulprüfungen):	keine
Angestrebte Lernergebnisse:	• Die Studierenden kennen den grundlegenden Aufbau und die Funktionsweise der Hardware von Rechnern. • Die Studierenden sind in der Lage, grundlegende Abhängigkeiten zwischen der Performanz von Software und Hardware zu verstehen. • Die Studierenden sind in der Lage, die Weiterentwicklung der relevanten Hardware in Ihrem beruflichen Umfeld zu verstehen und einzuordnen.
Inhalt:	• Geschichtliches, u.a. Mooresche Gesetz, Prozessor-Generationen • Rechner: Komponenten und Struktur , Funktionsweise , Buskommunikation, PC-Systeme • Logikbausteine: Kombinatorische und sequentielle Logik , Taktverfahren , Entwurf sequentieller Bausteine, Entwurf einer einfachen ALU • Prozessoren, RISC-Architektur vs. CISC-Architektur, Befehlssatzarchitekturen, Rechenleistung von Prozessoren, Pipelining

	• Speicher: Speichertechnologien, Speicherhierarchie, Hauptspeicher, Cachespeicher • Einfache Assemblerbeispiele als Brückenschlag zur Software
Studien- / Prüfungsleistungen:	Studienleistungen: Die Studierenden können während des Praktikums Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der Vorlesung bekannt gegeben. Prüfungsleistungen: Klausur
Literatur:	• Häberlein, Tobias, Technische Informatik Vieweg und Teubner Verlag, aktuellste Auflage • Hoffmann, Dirk W. , Grundlagen der Technischen Informatik, Hanser Verlag, aktuellste Auflage • Eventuelle weitere aktuelle Literatur wird im zugehörigen Moodle Kurs genannt
Bemerkungen:	-

Theoretische Informatik

Kürzel:	THI
Untertitel:	---
Studiensemester:	2. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Ulrike Griefahn
Dozent(in):	Prof. Dr. Ulrike Griefahn
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: 2 I/TI: 2 MI: 2 WI: -
Lehrform / SWS:	4 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30
Arbeitsaufwand:	Präsenzstudium: 67 Zeitstunden Eigenstudium: 113 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul, Einteilung der Übungsgruppen erfolgt nach der 1. Vorlesungsstunde über den Moodle-Kurs
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Logik und diskrete Strukturen
Angestrebte Lernergebnisse:	Die Studierenden können mit den wesentlichen Grundbegriffen der theoretischen Informatik umgehen. Sie sind in der Lage, die Korrektheit einfacher Algorithmen nachzuweisen. Sie können die Komplexität einfacher Algorithmen formal herleiten und algorithmische Probleme hinsichtlich ihrer Laufzeitkomplexität in Klassen einteilen. Die Studierenden kennen unterschiedliche formale Berechnungsmodelle und sind in der Lage, einfache Probleme mit ihnen zu lösen. Sie sind in der Lage, formale Sprachen in Klassen einzuteilen und mit Hilfe von Regelwerken zu beschreiben sowie abstrakte Maschinenmodelle zu definieren, um formale Sprachen zu erkennen.

	Der Besuch dieses Moduls versetzt die Studierenden insgesamt in die Lage, in ihrer zukünftigen Praxis handhabbare Probleme von nicht mehr handhabbaren zu unterscheiden, und bei der Lösung praktischer Probleme die Anwendbarkeit formaler Konzepte zu erkennen und diese einzusetzen.
Inhalt:	• Programmverifikation • Komplexität und Komplexitätsklassen • Berechenbarkeit und Berechnungsmodelle • Endliche Automaten und reguläre Sprachen • Kontextfreie Sprachen und Kellerautomaten • Chomsky-Hierarchie
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.)
Literatur:	Dirk W. Hoffmann, Theoretische Informatik, Carl Hanser Verlag, 3. Auflage, 2015, 432 Seiten, ISBN-13: 978-3446444461 Uwe Schöning: Theoretische Informatik – kurzgefasst, Spektrum Akademischer Verlag, 5. Auflage, 2003, 190 Seiten, ISBN-13: 978-3827418241
Bemerkungen:	---

Theoretische Informatik

Kürzel:	THI		
Untertitel:	---		
Studiensemester:	2. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr. Ulrike Griefahn		
Dozent(in):	Prof. Dr. Ulrike Griefahn		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	2	-	-
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung		
Gruppengröße:	Vorlesung: nicht begrenzt, Übung: 40		
Arbeitsaufwand:	Kontaktzeit: 56 Zeitstunden Selbststudium: 124 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Sommersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Vorlesung: keine, Übungen: über den Moodle-Kurs		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Logik und diskrete Strukturen, Grundlagen der Mathematik für Informatiker		
Angestrebte Lernergebnisse:	Die Studierenden können mit den wesentlichen Grundbegriffen der theoretischen Informatik umgehen. Sie sind in der Lage, die Korrektheit einfacher Algorithmen nachzuweisen. Sie können die Komplexität einfacher Algorithmen formal herleiten und algorithmische Probleme hinsichtlich ihrer Laufzeitkomplexität in Klassen einteilen. Die Studierenden kennen unterschiedliche formale Berechnungsmodelle und sind in der Lage, einfache Probleme mit ihnen zu lösen. Sie sind in der Lage, formale Sprachen in Klassen einzuteilen und mit Hilfe von Regelwerken zu beschreiben sowie abstrakte Maschinenmodelle zu definieren, um formale Sprachen zu erkennen. Der Besuch dieses Moduls versetzt die Studierenden insgesamt in die Lage, in ihrer zukünftigen Praxis handhabbare Probleme von nicht mehr handhabbaren		

	zu unterscheiden, und bei der Lösung praktischer Probleme die Anwendbarkeit formaler Konzepte zu erkennen und diese einzusetzen.
Inhalt:	• Programmverifikation • Komplexität und Komplexitätsklassen • Berechenbarkeit und Berechnungsmodelle • Formale Sprachen und Chomsky-Hierarchie • Endliche Automaten und reguläre Sprachen • Kontextfreie Sprachen und Kellerautomaten
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.)
Literatur:	• Dirk W. Hoffmann, Theoretische Informatik, Carl Hanser Verlag, 5. aktualisierte Auflage, 2022, 432 Seiten, ISBN: 978-3-446-47029-3 • Uwe Schöning: Theoretische Informatik – kurzgefasst, Spektrum Akademischer Verlag, 5. Auflage, 2003, 190 Seiten, ISBN-13: 978-3-827-41824-1
Bemerkungen:	---

Wahlpflichtkatalog Praktische Informatik

Betrieb komplexer verteilter Systeme

Kürzel:	BKV			
Untertitel:	---			
Studiensemester:	4. (Bachelor)			
Modulverantwortliche(r):	Prof. Dr. Andreas Cramer			
Dozent(in):	Prof. Dr. Andreas Cramer			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	WP	-	-	WP
Lehrform / SWS:	2 SWS Vorlesung, 2 SWS Praktikum			
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 54 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 126 Zeitstunden			
Leistungspunkte:	6			
Turnus:	Sommersemester, jährlich			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul			
Voraussetzungen nach Prüfungsordnung:	Keine			
Empfohlene Voraussetzungen (Modulprüfungen):	Betriebssysteme, Rechnernetze			
Angestrebte Lernergebnisse:	Die Studierenden lernen unterschiedliche Technologien und Konzepte kennen, die für den Betrieb großer IT-Infrastrukturen notwendig sind und bekommen erste praktische Erfahrungen mit deren Anwendung. Sie erlangen die Fähigkeit, neue Konzepte im Umfeld des IT-Betriebs schnell begreifen, einordnen und bewerten zu können.			
Inhalt:	Einführung Speichernetze Virtualisierung System-Management			
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine			

	Prüfungsleistungen: Klausur oder mündliche Prüfung
Literatur:	Bekanntgabe in der Vorlesung
Bemerkungen:	---

Betriebswirtschaftslehre für Informatiker

Kürzel:	BWIN
Untertitel:	Einführung in betriebswirtschaftliche Zusammenhänge anhand eines Unternehmensplanspiels.
Studiensemester:	5. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Siegbert Kern, Prof. Dr. Henning Ahlf
Dozent(in):	Prof. Dr. Siegbert Kern, Prof. Dr. Henning Ahlf
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP I/TI: WP MI: WP WI: -
Lehrform / SWS:	2 SWS Vorlesung, 2 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 56 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 104 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Keine Voraussetzungen
Angestrebte Lernergebnisse:	Die Studierende werden in die Lage versetzt: • die unternehmerischen Aufgaben der Betriebswirtschaftslehre zu verstehen und zu erläutern, • die wesentlichen Aufgaben der betrieblichen Funktionalbereiche und deren Interdependenzen zu verstehen, • die vermittelten betriebswirtschaftlichen Vorgehensweisen und Methoden anzuwenden.
Inhalt:	Vermittlung der wesentlichen betriebswirtschaftlichen Aufgaben und Methoden in einem Unternehmen. Rahmenbedingungen eines Unternehmens und Ziele eines Unternehmens, Unternehmensführung, Organisation, Marketing, Fertigung, Materialwirtschaft, Personal, Rechnungswesen, Finanzwirtschaft.

	Vertiefung der betriebswirtschaftlichen Inhalte mithilfe der Durchführung eines Planspiels mit TopSim.
Studien- / Prüfungsleistungen:	Die Teilnehmer der Veranstaltung nehmen am Planspiel und den dafür erforderlichen Präsenzterminen teil. Jeder Teilnehmer erläutert in Form einer Präsentation die getroffenen Unternehmensentscheidungen und deren Auswirkungen. Prüfungsleistung: Die erfolgreiche Teilnahme am Planspiel und die Präsentation werden als Prüfungsleistung bewertet.
Literatur:	Unterlagen zum TopSim Planspiel Olfert, K.; Rahn, H.-J.: Einführung in die Betriebswirtschaftslehre, 11. Auflage, Herne 2013. Wöhe, Günter; Döhring, Ulrich: Einführung in die Allgemeine Betriebswirtschaftslehre, 25. Auflage, München 2013.
Bemerkungen:	Mindestteilnehmerzahl 12

Bildverarbeitung

Kürzel:	BV			
Untertitel:	Grundlegende Verfahren und Algorithmen der digitalen Bildverarbeitung			
Studiensemester:	5. (Bachelor)			
Modulverantwortliche(r):	Prof. Dr. Hartmut Surmann			
Dozent(in):	Prof. Dr. Hartmut Surmann			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	WP	5	WP	-
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum			
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden			
Leistungspunkte:	6			
Turnus:	Wintersemester, jährlich			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	keine			
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen			
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung, Grundlagen der Mathematik für Informatiker, Lineare Algebra			
Angestrebte Lernergebnisse:	Die Studierenden kennen die Begriffe und Verfahren der digitalen Bildverarbeitung sowie die Konzepte und Methoden der Programmierung und können diese effektiv und strukturiert bei der Entwicklung eigener Bildverarbeitungsprogramme einsetzen. Lehrsprachen sind C / C++, Python. Bibliotheken: OpenCV, scikit-image			
Inhalt:	• Grundlagen / Begriffsbildung • Kameras • Bildverarbeitungsoperationen • Morphologische Operationen • Bildsegmentierung			

	• Merkmale von Objekten • Klassifikation • Bildverarbeitungssoftware
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• J. Steinmüller: „Bildanalyse“, Springer Verlag, ISBN 978-3-540-79743-2. • A. Nischwitz, P. Haberäcker: „Computergrafik und Bildverarbeitung, Band II Bildverarbeitung“, Teubner Verlag, ISBN 978-3-834-81712-9. • A Kaehler, G. Bradski: „Computer Vision in C++ with the OpenCV Library“, O'Reilly, ISBN 978-1-449-31465-1.
Bemerkungen:	---

Grundlagen der IT-Sicherheit

Kürzel:	ITS
Untertitel:	Grundlegende Konzepte, Architekturen, Prinzipien und Funktionsweisen von Sicherheitskomponenten und -systemen
Studiensemester:	5. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. (TU NN) Norbert Pohlmann
Dozent(in):	Prof. Dr. (TU NN) Norbert Pohlmann
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: WP; MI: WP; WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 54 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 126 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Erfolgreich absolviertes Praktikum
Empfohlene Voraussetzungen (Modulprüfungen):	Keine
Angestrebte Lernergebnisse:	• Gutes Verständnis von möglichen Angriffen und geeigneten Gegenmaßnahmen in der IT • Erlangen von Kenntnissen über den Aufbau, die Prinzipien, die Architektur und die Funktionsweise von Sicherheitskomponenten und -systemen • Sammeln von Erfahrungen bei der Ausarbeitung und Präsentation von neuen Themen aus dem Bereich IT-Sicherheit • Gewinnen von praktischen Erfahrungen über die Nutzung und die Wirkung von Sicherheitssystemen • Erleben der Notwendigkeit und Wichtigkeit der IT-Sicherheit
Inhalt:	• Einführung: IT-Sicherheit als Wirkungs- und Handlungszusammenhang,

	Sicherheitsbedürfnisse, Bedrohungen, Angriffe, Schadenskategorien, Eintrittswahrscheinlichkeiten • Kryptographie und technologische Grundlagen für Schutzmaßnahmen: Private-Key-Verfahren, Public-Key-Verfahren, Kryptoanalyse, Hashfunktionen, Schlüsselgenerierung • Sicherheitsmodule (SmartCards, TPM, high-security und high-performence Lösungen) • Authentikationsverfahren: Grundsätzliche Prinzipien sowie unterschiedliche Algorithmen und Verfahren • ID-Management (Idee, Ziel, Konzepte) • ID-Cards (Neuer Personalausweis) • IT und Internet Frühwarnsysteme (Grundlagen) • Trusted Computing (Grundlagen)
Studien- / Prüfungsleistungen:	Studienleistungen: Erfolgreich absolviertes Praktikum Prüfungsleistungen: Klausur (90 Min.)
Literatur:	Pohlmann, N.: Firewall-Systeme - Sicherheit für Internet und Intranet, E- Mail-Security, Virtual Private Network, Intrution Detection-System, Personal Firewalls. 5. aktualisierte und erweiterte Auflage; ISBN 3- 8266-0988-3; MITP-Verlag, Bonn 2003 A Campo, M.; Pohlmann, N.: Virtual Private Network (VPN). 2. aktualisierte und erweiterte Auflage, ISBN 3-8266-0882-8; MITP-Verlag, Bonn 2003 Pohlmann, N.; Reimer, H.: "Trusted Computing - Ein Weg zu neuen IT- Sicherheitsarchitekturen”, ISBN 978-3-8348-0309-2, Vieweg-Verlag, Wiesbaden 2008 H. Blumberg, N. Pohlmann: "Der IT-Sicherheitsleitfaden“, 2. aktualisierte und erweiterte Auflage, ISBN-10: 3-8266-1635-9; 523 Seiten, MITP-Verlag, Bonn 2006 M. Hertlein, P. Manaras, N. Pohlmann: “Bring Your Own Device For Authentication (BYOD4A) – The Xign-System”. In Proceedings of the ISSE 2015 - Securing Electronic Business Processes - Highlights of the Information Security Solutions Europe 2015 Conference, Eds.: N. Pohlmann, H. Reimer, W. Schneider; Springer Vieweg Verlag, Wiesbaden 2015
Bemerkungen:	---

IT-Recht

Kürzel:	ITR			
Untertitel:	Rechtliche Aspekte bei der Erstellung und Anwendung von Softwareprodukten aller Art.			
Studiensemester:	5. (Bachelor)			
Modulverantwortliche(r):	Studiengangsbeauftragte/r Wirtschaftsinformatik			
Dozent(in):	Lehrbeauftragte/r			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	WP	WP	WP	5
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung			
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 56 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 104 Zeitstunden			
Leistungspunkte:	6			
Turnus:	Wintersemester, jährlich			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul			
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen			
Empfohlene Voraussetzungen (Modulprüfungen):	Keine Voraussetzungen			
Angestrebte Lernergebnisse:	Die Studierende werden in die Lage versetzt: die relevanten rechtlichen Aspekte und gesetzlichen Regelungen als Randbedingung in ihre berufliche Arbeit einbeziehen können, zu wissen, welche datenschutzrechtlichen Vorgaben es bei der Speicherung personenbezogener Daten gibt oder welche rechtlichen Regeln bei der Gestaltung und Programmierung von Internet-Auftritten einzuhalten sind.			
Inhalt:	Rechtliche Aspekte bei der Erstellung und Anwendung von Softwareprodukten aller Art, Internet-, Datenschutz- und Urheberrecht, die für die behandelten Rechtsfelder maßgeblichen europäischen und deutschen Gesetze.			

Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.)
Literatur:	Nach Bekanntgabe in der Vorlesung.
Bemerkungen:	---

Komponentenbasierte Softwareentwicklung

Kürzel:	KBE			
Untertitel:	Komponentenbasierte Softwareentwicklung und Frameworks			
Studiensemester:	5. (Wahlpflicht im Bachelor)			
Modulverantwortliche(r):	Prof. Jürgen Znotka			
Dozent(in):	Prof. Jürgen Znotka			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	WP	-	-	-
Lehrform / SWS:	2 SWS Vorlesung, 2 Praktikum			
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 53 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 97 Zeitstunden			
Leistungspunkte:	6			
Turnus:	Wintersemester, nach Bedarf			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul			
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen			
Empfohlene Voraussetzungen (Modulprüfungen):	Softwaretechnik, Datenbanken, Internet-Datenbanken, GUI-Programmierung			
Angestrebte Lernergebnisse:	Die Studierenden kennen			

	• Begriffe der komponentenbasierten Softwareentwicklung • Begriffe der speziellen JEE Entwicklung (Session Beans, Singleton, Message-Driven Beans) • Webservices • Begriffe im Kontext von Frameworks (Inversion of Control IoC) • Begriffe der Aspektorientierte Softwareentwicklung • die folgenden Diagramme der UML: Komponentendiagramm, Verteilungsdiagramm • Begriffe der Softwarequalität wie Functionality, Usability, Reliability, Portability und Supportability (FURPS) Die Studierenden verstehen • den Zusammenhang der einzelnen Phasen in verschiedenen Softwareprozessen und die jeweiligen Vor- und Nachteile • den Zusammenhang zwischen Anforderungen und objektorientierten Modellen • Die Studierenden können das Erlernte anwenden, um • aus unstrukturierten Anforderungen an ein System funktionale Anforderungen zu extrahieren • qualitative Anforderungen zu formulieren • objektorientierte Modelle auf Basis der UML zu erstellen für verschiedene Anwendungsdomänen
Inhalt:	• Einführung komponentenbasierte Softwareentwicklung • Java Enterprise Komponentenmodell • Session Beans • Singleton Bean • Message-Driven Beans • Webservices • Aspektorientierte Softwareentwicklung • Einführung in Frameworks • Ein spezielles Framework • UML Diagramme: Komponentendiagramm und Verteilungsdiagramm
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (60 Min.)

Literatur:	Sommerville, Ian: Software Engineering, Addison-Wesley, 10th Edition, 2015 George T. Heineman, William T. Councill: Component-Based Software Engineering: Putting the Pieces Together, Addison-Wesley Professional, 2001 Clemens Szyperski: Component Software: Beyond Object-Oriented Programming. 2nd Edition, Addison-Wesley, 2002 Eric Jendrock, Ricardo Cervera-Navarro, Ian Evans, Kim Haase, William Markito: The Java EE 7 Tutorial, 2014 SPRING Framework documentation: https://spring.io/
Bemerkungen:	---

Künstliche Intelligenz

Kürzel:	KI
Untertitel:	---
Studiensemester:	4. (Wahlpflicht im Bachelor)
Modulverantwortliche(r):	Prof. Dr. Wolfram Conen
Dozent(in):	Prof. Dr. Wolfram Conen
Sprache:	
Zuordnung zum Curriculum:	I/PI: WP; I/TI: -; MI: -; WI: -
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 54 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 126 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, unregelmäßig
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Logik und diskrete Strukturen, Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung
Angestrebte Lernergebnisse:	Die Studierenden kennen die Grundzüge der Entwicklungsgeschichte der Künstlichen Intelligenz (KI). Sie kennen grundlegende Begriffe der Stochastik und des maschinellen Lernens, insbesondere der bayes'schen Modellierung, und können diese anwenden. Sie sind in der Lage, typische Problemsituationen aus den Feldern intelligentes Datamining (Klassifikation, Lernen aus Daten, Bayes'sche Inferenz) und Optimierung rationaler Entscheidungen (insbesondere Planen und Entscheiden bei unsicherem Wissen) zu modellieren und zu lösen. Sie können ihre Erkenntnisse auf verwandte Problemstellungen übertragen und sind darauf

	vorbereitet, sich vertieft mit Spezialgebieten der KI-Anwendung auseinanderzusetzen.
Inhalt:	Grundbegriffe der Stochastik (Zufallsvariablen, Verteilungen, Bayes'sches Gesetz, Schätzen und Testen). Modellierung und Anwendung von Wissen bei Unsicherheit: Bayes'sche Inferenz, Sampling, Filtering, Decision Making. Maschinelles Lernen aus Daten und Texten: Kategorisierung (Naive Bayes, kNN, Decision Trees), Clustering, Collaborative Filtering, Time Series Analysis. Musterkennung in diskreten Datenströmen (NN, stochastische Methoden) oder ein aktuelles Thema zu Data Mining.
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Klausur (90 Min.)
Literatur:	Skript, ergänzend: Russell, Norvig: Artificial Intelligence, A Modern Approach, Pearson Marsland: Machine Learning, An Algorithmic Perspective, CRC Press jeweils in aktueller Auflage, und aktuelle Artikel.
Bemerkungen:	---

Mobile Computing

Kürzel:	MOC			
Untertitel:	Einführung in Bluetooth, WLAN, Mobilfunknetze (zweite bis vierte Generation)			
Studiensemester:	4. (Wahlpflicht im Bachelor)			
Modulverantwortliche(r):	Prof. Dr. Detlef Mansel			
Dozent(in):	Prof. Dr. Detlef Mansel			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	WP	-	-	WP
Lehrform / SWS:	2 SWS Vorlesung, 2 SWS Übung, 1 SWS Praktikum			
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden			
Leistungspunkte:	6			
Turnus:	Sommersemester, unregelmäßig			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	Keine, die Einteilung der Übungs- und Praktikumsgruppen wird in der 1. Vorlesungsstunde besprochen.			
Voraussetzungen nach Prüfungsordnung:	Keine			
Empfohlene Voraussetzungen (Modulprüfungen):	Rechnernetze			
Angestrebte Lernergebnisse:	Die Studierenden kennen den grundlegenden Aufbau und die Systemeigenschaften der für mobile Anwendungen verwendeten Mobilfunksysteme. Sie können grundlegend mit den Einschränkungen der Funkanbindung mobiler Endgeräte umgehen und wissen bis zu einem gewissen Grade, welchen Einfluss diese Einschränkungen auf die Effizienz der Programme haben.			
Inhalt:	Typen mobiler Netze• Bluetooth als Beispiel für ein Ad hoc Netz• GSM/UMTSLTE als zellulares Infrastruktur-Netz• Wireless LAN (WLAN) Grundlagen• Eigenschaften mobiler Endgeräte•			
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistung einen			

	Bonus im Einklang mit §11 der Rahmenprüfungsordnung: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss im Funk-Labor gelöst und dokumentiert werden (Protokolle). Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (60 Min.)
Literatur:	Michael Sauter , Grundkurs Mobile Kommunikationssysteme, Springer Vieweg, ISBN 978-3-658-08341-0, 2015 Aktuelle Ergänzungen auf den Moodle-Kurs zu diesem Modul
Bemerkungen:	---

Mobile Robotik

Kürzel:	MRO
Untertitel:	Grundlegende Konzepte und Verfahren zur Entwicklung von Applikationen für mobile Roboter
Studiensemester:	5. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Hartmut Surmann
Dozent(in):	Prof. Dr. Hartmut Surmann
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: 5; MI: -; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	keine
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung, Grundlagen der Mathematik für Informatiker, Lineare Algebra
Angestrebte Lernergebnisse:	Die Studierenden kennen die Begriffe und Komponenten von mobilen Robotern sowie die Konzepte und Methoden der Programmierung und können diese effektiv und strukturiert bei der Entwicklung eigener Steuerungsprogramme einsetzen. Sie wissen wie unterschiedliche Sensordaten fusioniert werden und mobile Systeme navigieren und sich selbst lokalisieren. Sie kennen die Gefahren beim Umgang mit mobilen Systemen und die Wichtigkeit der Einhaltung von Vorschriften sowohl auf technischer als auch sozialer Ebene. Die Studierenden sind zudem in der Lage, sich selbstständig und zügig in unterschiedliche Arten von mobilen Robotern und deren Programmierumgebung einzuarbeiten. Betriebssystem: Linux + ROS, Lehrsprache ist C / C++, Python.

Inhalt:	Einführung / Begriffsbildung Mobile Roboter • Sensorik • Aktuatorik • Lokalisierung • Kartierung • Navigation • Planung
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• J. Hertzberg, K. Lingemann, A. Nüchter: „Mobile Roboter: Eine Einführung aus Sicht der Informatik“, ISBN 978-3642017254 • B. Siciliano, O. Khatib (Eds.): „Handbook of Robotic“, ISBN 978-3-540-23957-4 • Craig, J.J. (2004), „Introduction to Robotics: Mechanics and Control (3rd Edition)“, 8, 2004. Prentice Hall • R. Siegwart „Introduction to Autonomous Mobile Robots“, MIT Press, ISBN: 978-0-262-19502 -7 • S. Thrun, W. Burgard, D. Fox: „Probabilistic Robotics“, ISBN 978-0262201629
Bemerkungen:	---

Objektorientierte Programmierung mit C++

Kürzel:	OPC
Untertitel:	---
Studiensemester:	5. (Wahlpflicht im Bachelor)
Modulverantwortliche(r):	Studiengangsbeauftragte/r Informatik
Dozent(in):	Prof. Dr. Reinhard Wierich oder Lehrbeauftragte/r
Sprache:	Deutsch; bei englischsprachigen Teilnehmern Englisch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: WP; MI: WP; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Standard
Arbeitsaufwand:	Standard
Leistungspunkte:	6
Turnus:	Wintersemester, unregelmäßig
Teilnehmerzahl:	Standard
Anmeldungsmodalitäten:	Siehe Aushang
Voraussetzungen nach Prüfungsordnung:	Erfolgreich absolviertes Praktikum
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung
Angestrebte Lernergebnisse:	• Verständnis für die Konzepte der Objektorientierten Programmierung in C++ • Sicherer Umgang mit der Syntax von C++, speziell Pointern und Referenzen • Verständnis des Operator Overloading – Konzepts, speziell in Verbindung mit Containern • Eingeübter Umgang mit Templates und entsprechenden Bibliotheken
Inhalt:	• Syntaxerweiterungen gegenüber C • Konzeptuelle Unterschiede gegenüber JAVA • Default- und Kopier-Konstruktor, Initialisierungsliste • Operator Overloading • Vererbung und Mehrfachvererbung • Templates, Container und Iteratoren • Arbeiten mit der Standard Template Library • Fortgeschrittene Template-Anwendungen • Exception Handling in C++
Studien- / Prüfungsleistungen:	Studienleistungen: Erfolgreich absolviertes Praktikum Prüfungsleistungen: Klausur (120 Min.)

Literatur:	Prata, Stephen: C++**.** te-wi Verlag, ISBN 3-89362-701-4. Prinz, P. *et al*.,: Objektorientiert programmieren mit ANSI C++. Verlag Prentice Hall, 1998, ISBN 3-8272-9560-2. Stroustrups, Die C++ Programmiersprache. Addison-Wesley, 2000. Koenig *et al*., Intensivkurs C++. Addison-Wesley, 2003, ISBN 3-8273-7029-9
Bemerkungen:	---

Parallele Programmierung

Kürzel:	PAP			
Untertitel:	Grundlegende Methoden und Algorithmen paralleler Programmierung			
Studiensemester:	4. (Wahlpflicht im Bachelor)			
Modulverantwortliche(r):	Prof. Dr. Hartmut Surmann			
Dozent(in):	Prof. Dr. Hartmut Surmann			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	WP	WP	WP	-
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum			
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden			
Leistungspunkte:	6			
Turnus:	Sommersemester, jährlich			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	Keine			
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen			
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung, Grundlagen der Mathematik für Informatiker			
Angestrebte Lernergebnisse:	Die Studierenden lernen die Grundlagen der Parallelen Programmierung kennen und können parallele Programme entwickeln und testen. Weiterhin lernen sie klassische Algorithmen zu parallelisieren und innerhalb eines Rechners und über mehrere Rechner hinweg parallel zu verteilen. Neben der Programmiermethodik und dem Design lernen die Studierenden die speziellen Probleme und Fragestellungen bei der parallelen Programmierung kennen, insbesondere die schwierigere Fehleranalyse.			
Inhalt:	Mit dem Aufkommen von leistungsfähigen Grafikkarten, Multicore- Prozessoren in PCs und Smartphones erhält die Parallel Programmierung einen immer höheren Stellenwert und ist für eine Vielzahl von Anwendungen			

	interessant. Die Vorlesung deckt die nachfolgenden Themengebieten ab: • Grundlagen paralleler Programmierung • Parallele Architekturen • Design paralleler Algorithmen • Threads • OpenMP • MPI • OpenCL • CUDA
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• Thomas Rauber: “Parallele Programmierung”, Springer Verlag, ISBN 978-3-540-46549-2. • D. E. Culler, J. Pal Singh, A. Gupta: “Parallel Computer Architecture: A Hardware/Software Approach”, Morgan Kaufmann, ISBN 1-55860-343-3. • I. Foster. ”Designing and Building Parallel Programs”, Addison Wesley, ISBN 978-0-201-57594-1. • P. Pacheco: “An Introduction to Parallel Programming”, Morgan Kaufmann, ISBN 978-0-123-74260-5 • Patterson, Hennessy, Bode: “Rechnerorganisation und –entwurf”, Standardwerk, Spektrum Akademischer Verlag, ISBN 978-3-827-41595-0 .
Bemerkungen:	---

Practical Security Attacks and Exploitation

Kürzel:	PRAX
Untertitel:	Practical Security Attacks and Exploitation
Studiensemester:	4. oder 5. (Wahlpflicht im Bachelor)
Modulverantwortliche(r):	Prof. Dr. Christian Dietrich
Dozent(in):	Prof. Dr. Christian Dietrich
Sprache:	Englisch oder deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: WP; MI: WP; WI: WP
Lehrform / SWS:	1 SWS Vorlesung, 3 SWS Praktikum
Gruppengröße:	Vorlesung: nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 52 h Eigenstudium inkl. Projekt/Ausarbeitung und Prüfungsvorbereitung: 128 h
Leistungspunkte:	6
Turnus:	Sommer- und Wintersemester, unregelmäßig
Teilnehmerzahl:	Keine Beschränkung
Anmeldungsmodalitäten:	Anmeldung via Moodle
Voraussetzungen nach Prüfungsordnung:	Keine besonderen
Empfohlene Voraussetzungen (Modulprüfungen):	Grundlegende IT-Sicherheitskenntnisse
Angestrebte Lernergebnisse:	• Verständnis gängiger Verfahren zur Systemsicherheit, Systemintegrität und zum Softwareschutz • Anwenden von Mechanismen zur Identifikation und Ausnutzung von Software-Schwachstellen • Anwenden von Angriffstechniken in Computernetzwerken • Erlangen von Kenntnissen im Bereich der Schadsoftware-Erkennung und -Abwehr • Teilnahme an einem Capture-the-Flag-Wettbewerb (z.B. Cyber Security Challenge)
Inhalt:	Die Studierenden lernen die Anwendbarkeit und Grenzen von sicherheitsrelevanten Angriffen gegen Systeme, Netzwerkprotokolle und Software. Dabei werden die folgenden Themen behandelt: • Introduction to vulnerability research • Reconnaissance and scanning • Infection vectors

	• System security and operational security • Software security • Network security • Denial-of-Service attacks • Web security • Incident response Lerneinheiten bestehen jeweils aus einer Einführung in Form mindestens einer Vorlesungseinheit sowie Aufgaben, die im Praktikum gelöst werden müssen. Darüber hinaus müssen die Studierenden selbst verwundbare Beispiele als Aufgaben entwerfen, die beispielsweise im Rahmen eines eigenen CTF-Wettbewerbs eingesetzt werden könnten.
Studien- / Prüfungsleistungen:	Studienleistungen: Die Studierenden können während des Praktikums Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der Vorlesung bekannt gegeben. Prüfungsleistungen: Ausarbeitung sowie Klausur oder mündliche Prüfung
Literatur:	Eckert, C.: *IT-Sicherheit. Konzepte, Verfahren, Protokolle.* Oldenbourg, München, aktuellste Auflage Erickson, J.: *Hacking - The Art of Exploitation*. No Starch Press; aktuellste Auflage Aktuelle wissenschaftliche Publikationen
Bemerkungen:	---

Robotik

Kürzel:	ROB			
Untertitel:	Grundlegende Konzepte und Verfahren zur Entwicklung von Applikationen für Roboter / Manipulatoren			
Studiensemester:	4. (Bachelor)			
Modulverantwortliche(r):	Prof. Dr. Hartmut Surmann			
Dozent(in):	Prof. Dr. Hartmut Surmann			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	WP	4	WP	-
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum			
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden			
Leistungspunkte:	6			
Turnus:	Sommersemester, jährlich			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	keine			
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen			
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung, Grundlagen der Mathematik für Informatiker, Lineare Algebra			
Angestrebte Lernergebnisse:	Die Studierenden kennen die Begriffe und Komponenten von Industrierobotern sowie die Konzepte und Methoden der Programmierung und können diese effektiv und strukturiert bei der Entwicklung eigener Steuerungsprogramme einsetzen. Sie gehen sicher mit maschinennahen Steuerungen und Regelungen um und wissen, welchen Einfluss die Steuerung und Regelung auf das Roboterverhalten hat. Sie kennen die Gefahren und Wichtigkeit der Einhaltung von Vorschriften sowohl auf technischer als auch sozialer Ebene beim Umgang mit Manipulatoren. Die Studierenden sind zudem in der Lage, sich selbstständig und zügig in unterschiedliche Industrieroboter und deren Programmierumgebung			

	einzuarbeiten. Lehrsprachen sind KUKA Robot Programming Language, C / C++, Python.
Inhalt:	Grundlagen der Industrierobotik / Manipulatortechnik •Begriffsbildung Komponenten eines Manipulators / Industrieroboters • Beschreibung einer Roboterstellung • Transformation zwischen Roboter- und Weltkoordinaten • Bewegungsart und Interpolation • Roboterprogrammierung
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• Wolfgang Weber: Industrieroboter, Methoden der Steuerung und Regelung, Hanser Verlag, 2. Auflage, ISBN 978-3-446-41031-2 • Stefan Hesse: Grundlagen der Handhabungstechnik, Hanser Verlag, ISBN 978-3-446-40473-1 • Bruno Siciliano, Oussama Khatib (Eds.): Handbook of Robotic, ISBN 978-3-540-23957-4 • Craig, J.J. (2004), "Introduction to Robotics: Mechanics and Control (3rd Edition)", 8, 2004. Prentice Hall
Bemerkungen:	---

Software Design

Kürzel:	SWD
Untertitel:	Architektur und Design komplexer Softwaresysteme
Studiensemester:	4. (Wahlpflicht im Bachelor)
Modulverantwortliche(r):	Prof. Jürgen Znotka
Dozent(in):	Prof. Jürgen Znotka
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: -; MI: -; WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 55 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 95 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, nach Bedarf
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Softwaretechnik, Datenbanken, Internet-Datenbanken, GUI-Programmierung
Angestrebte Lernergebnisse:	Die Studierenden kennen • Architekturmuster • Designmuster • OSGi Komponentenmodell Die Studierenden verstehen • den Zusammenhang der einzelnen Phasen in verschiedenen Softwareprozessen und die jeweiligen Vor- und Nachteile, insbesondere den Übergang von Analyse zu Design Die Studierenden können das Erlernte anwenden, um • aus einem Pflichtenheft ein Design zu entwickeln • qualitative Anforderungen an das Design zu formulieren • objektorientierte Designmodelle auf Basis der UML zu erstellen

Inhalt:	• Einführung Software Design • Design Patterns (Observer, Adapter, Fassade, Strategie, Dekorierer, Simple Fabrik, Fabrikmethode, abstrakte Fabrik, Watchdog) • Einführung in Architekturmuster • MVC (Model-View-Controller) und dessen Derivate Passive View und Supervising Controller • Mehrschichtarchitektur • UML Diagramme: Interaktionsübersicht, Kommunikationsdiagramm, Paketdiagramm, Kompositionsstrukturdiagramm, Komponentendiagramm, Verteilungsdiagramm) • Komponentenmodell OSGi
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (60 Min.)
Literatur:	Sommerville, Ian: Software Engineering, Addison-Wesley, 10th Edition, 2015 Fowler, Martin: Patterns of Enterprise Application Architecture, Addison-Wesley, 2002 Rup, Chris u.a. UML 2 glasklar: Praxiswissen für die UML-Modellierung, Hanser, 4. Auflage, 2012 Kirk Knoernschild: Java Application Architecture: Modularity Patterns with Examples Using OSGi, Prentice Hall, 2012
Bemerkungen:	---

Wahlpflichtkatalog Technische Informatik

Betriebswirtschaftslehre für Informatiker

Kürzel:	BWIN
Untertitel:	Einführung in betriebswirtschaftliche Zusammenhänge anhand eines Unternehmensplanspiels.
Studiensemester:	5. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Siegbert Kern, Prof. Dr. Henning Ahlf
Dozent(in):	Prof. Dr. Siegbert Kern, Prof. Dr. Henning Ahlf
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: WP; MI: WP; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 2 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 56 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 104 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Keine Voraussetzungen
Angestrebte Lernergebnisse:	Die Studierende werden in die Lage versetzt: • die unternehmerischen Aufgaben der Betriebswirtschaftslehre zu verstehen und zu erläutern, • die wesentlichen Aufgaben der betrieblichen Funktionalbereiche und deren Interdependenzen zu verstehen, • die vermittelten betriebswirtschaftlichen Vorgehensweisen und Methoden anzuwenden.
Inhalt:	Vermittlung der wesentlichen betriebswirtschaftlichen Aufgaben und Methoden in einem Unternehmen. Rahmenbedingungen eines Unternehmens und Ziele eines Unternehmens, Unternehmensführung,

	Organisation, Marketing, Fertigung, Materialwirtschaft, Personal, Rechnungswesen, Finanzwirtschaft. Vertiefung der betriebswirtschaftlichen Inhalte mithilfe der Durchführung eines Planspiels mit TopSim.
Studien- / Prüfungsleistungen:	Die Teilnehmer der Veranstaltung nehmen am Planspiel und den dafür erforderlichen Präsenzterminen teil. Jeder Teilnehmer erläutert in Form einer Präsentation die getroffenen Unternehmensentscheidungen und deren Auswirkungen. Prüfungsleistung: Die erfolgreiche Teilnahme am Planspiel und die Präsentation werden als Prüfungsleistung bewertet.
Literatur:	Unterlagen zum TopSim Planspiel Olfert, K.; Rahn, H.-J.: Einführung in die Betriebswirtschaftslehre, 11. Auflage, Herne 2013. Wöhe, Günter; Döhring, Ulrich: Einführung in die Allgemeine Betriebswirtschaftslehre, 25. Auflage, München 2013.
Bemerkungen:	Mindestteilnehmerzahl 12

Grundlagen der IT-Sicherheit

Kürzel:	ITS
Untertitel:	Grundlegende Konzepte, Architekturen, Prinzipien und Funktionsweisen von Sicherheitskomponenten und -systemen
Studiensemester:	5. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. (TU NN) Norbert Pohlmann
Dozent(in):	Prof. Dr. (TU NN) Norbert Pohlmann
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: WP; MI: WP; WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 54 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 126 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Erfolgreich absolviertes Praktikum
Empfohlene Voraussetzungen (Modulprüfungen):	Keine
Angestrebte Lernergebnisse:	• Gutes Verständnis von möglichen Angriffen und geeigneten Gegenmaßnahmen in der IT • Erlangen von Kenntnissen über den Aufbau, die Prinzipien, die Architektur und die Funktionsweise von Sicherheitskomponenten und -systemen • Sammeln von Erfahrungen bei der Ausarbeitung und Präsentation von neuen Themen aus dem Bereich IT-Sicherheit • Gewinnen von praktischen Erfahrungen über die Nutzung und die Wirkung von Sicherheitssystemen • Erleben der Notwendigkeit und Wichtigkeit der IT-Sicherheit
Inhalt:	• Einführung: IT-Sicherheit als Wirkungs- und Handlungszusammenhang,

	Sicherheitsbedürfnisse, Bedrohungen, Angriffe, Schadenskategorien, Eintrittswahrscheinlichkeiten • Kryptographie und technologische Grundlagen für Schutzmaßnahmen: Private-Key-Verfahren, Public-Key-Verfahren, Kryptoanalyse, Hashfunktionen, Schlüsselgenerierung • Sicherheitsmodule (SmartCards, TPM, high-security und high-performence Lösungen) • Authentikationsverfahren: Grundsätzliche Prinzipien sowie unterschiedliche Algorithmen und Verfahren • ID-Management (Idee, Ziel, Konzepte) • ID-Cards (Neuer Personalausweis) • IT und Internet Frühwarnsysteme (Grundlagen) • Trusted Computing (Grundlagen)
Studien- / Prüfungsleistungen:	Studienleistungen: Erfolgreich absolviertes Praktikum Prüfungsleistungen: Klausur (90 Min.)
Literatur:	Pohlmann, N.: Firewall-Systeme - Sicherheit für Internet und Intranet, E- Mail-Security, Virtual Private Network, Intrution Detection-System, Personal Firewalls. 5. aktualisierte und erweiterte Auflage; ISBN 3- 8266-0988-3; MITP-Verlag, Bonn 2003 A Campo, M.; Pohlmann, N.: Virtual Private Network (VPN). 2. aktualisierte und erweiterte Auflage, ISBN 3-8266-0882-8; MITP-Verlag, Bonn 2003 Pohlmann, N.; Reimer, H.: "Trusted Computing - Ein Weg zu neuen IT- Sicherheitsarchitekturen”, ISBN 978-3-8348-0309-2, Vieweg-Verlag, Wiesbaden 2008 H. Blumberg, N. Pohlmann: "Der IT-Sicherheitsleitfaden“, 2. aktualisierte und erweiterte Auflage, ISBN-10: 3-8266-1635-9; 523 Seiten, MITP-Verlag, Bonn 2006 M. Hertlein, P. Manaras, N. Pohlmann: “Bring Your Own Device For Authentication (BYOD4A) – The Xign-System”. In Proceedings of the ISSE 2015 - Securing Electronic Business Processes - Highlights of the Information Security Solutions Europe 2015 Conference, Eds.: N. Pohlmann, H. Reimer, W. Schneider; Springer Vieweg Verlag, Wiesbaden 2015
Bemerkungen:	---

IT-Recht

Kürzel:	ITR
Untertitel:	Rechtliche Aspekte bei der Erstellung und Anwendung von Softwareprodukten aller Art.
Studiensemester:	5. (Bachelor)
Modulverantwortliche(r):	Studiengangsbeauftragte/r Wirtschaftsinformatik
Dozent(in):	Lehrbeauftragte/r
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: WP; MI: WP; WI: 5
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 56 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 104 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Keine Voraussetzungen
Angestrebte Lernergebnisse:	Die Studierende werden in die Lage versetzt: die relevanten rechtlichen Aspekte und gesetzlichen Regelungen als Randbedingung in ihre berufliche Arbeit einbeziehen können, zu wissen, welche datenschutzrechtlichen Vorgaben es bei der Speicherung personenbezogener Daten gibt oder welche rechtlichen Regeln bei der Gestaltung und Programmierung von Internet-Auftritten einzuhalten sind.
Inhalt:	Rechtliche Aspekte bei der Erstellung und Anwendung von Softwareprodukten aller Art, Internet-, Datenschutz- und Urheberrecht, die für die behandelten Rechtsfelder maßgeblichen europäischen und deutschen Gesetze.

Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.)
Literatur:	Nach Bekanntgabe in der Vorlesung.
Bemerkungen:	---

Medientechnik

Kürzel:	MET			
Untertitel:	Einführung in die Konzepte und Verfahren digitaler Audio-, Bild- und Videotechnik			
Studiensemester:	5. (Bachelor)			
Modulverantwortliche(r):	Prof. Dr. Ekkehard Schrey			
Dozent(in):	Prof. Dr. Ekkehard Schrey			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	-	WP	5	-
Lehrform / SWS:	3 SWS Vorlesung, 2 SWS Übung			
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 72 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 108 Zeitstunden			
Leistungspunkte:	6			
Turnus:	Wintersemester, jährlich			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul			
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen			
Empfohlene Voraussetzungen (Modulprüfungen):	Grundlagen der Mathematik für Informatiker, Mathematik für Medieninformatiker, bzw. Informatiker Technische Grundlagen der Informatik, Algorithmen und Datenstrukturen, Einführung in die Programmierung			
Angestrebte Lernergebnisse:	Die Studierenden • erlangen Kenntnisse der physikalischen, biologischen und wahrnehmungspsychologischen Grundlagen für Aufnahme, Übertragung und Wiedergabe von Audio-, Bild- und Video- Signalen und sind in der Lage, die Kenntnisse zur Lösung von konkreten Aufgabenstellungen selbständig anzuwenden • verstehen die Zusammenhänge von Zeit- und Frequenzfunktionen und sind in der Lage, diese			

	bei konkreten Aufgabenstellungen selbständig anzuwenden • erlangen Kenntnisse der Qualitätsmerkmale und Anforderungen an Audio- und Video-Systeme und können diese zur Lösung von konkreten Aufgabenstellungen selbständig nutzen • erlangen Kenntnisse über Grundlagen und Konzepte zur Bearbeitung digitaler Audio-, Bild- und Videosignale und sind in der Lage, diese in eigenen Worten zu beschreiben • erhalten einen Überblick über die wichtigsten Standards im Audio-, Bild- und Videobereich und sind in der Lage, die Standards in eigenen Worten zu beschreiben
Inhalt:	• Grundlagen Akustik (Schallerzeugung, Übertragung und Empfang von Schall, Schallwahrnehmung, Kenngrößen von Schall) • Elektroakustische Wandler und ihre Qualitätsmerkmale • Beschreibung von Signalen und Systemen im Zeit– und Frequenzbereich • Digitalisierung: AD-/DA-Konverter. Abtastung, Quantisierung, Quantisierungsfehler, Beschreibung abgetasteter Signale im Zeit- und Frequenzbereich • Speicherung und Übertragung von Audiosignalen (analog, digital) • Bearbeitung digitaler Audio-Signale: Modifikation im Zeit- und Frequenzbereich • Natur des Lichts, Farbwahrnehmung, Farbstandards • Prinzipien optoelektrischer Wandler (Aufnahme, Wiedergabe) • Bild- und Videodatenformate, Videostandards • Bildbearbeitung: Modifikation von Helligkeit, Kontrast, Intensität und Ortskoordinaten • Videobearbeitung: Konzept des nichtlinearen Videoschnitts
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Vorrechnung einer gewissen Anzahl von Übungsaufgaben. Die Anzahl der Aufgaben wird zu Beginn der Veranstaltung so festgelegt, dass jeder Studierende diese Anzahl gut bewältigen kann. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.)

Literatur:	Raffaseder, H.: Audiodesign. Fachbuchverlag Leipzig, 2002, ISBN 3- 446-21828-9 Stotz, D.: Computerunterstützte Audio- und Videotechnik. Springerverlag, 2011, ISBN 978-3-642-23252-4 Heyna, A.Briede, M; Schmidt, U.: Datenformate im Medienbereich. Fachbuchverlag Leipzig, 2003, ISBN 3-446-22542-0 Henning, P. A.: Taschenbuch Multimedia. Fachbuchverlag Leipzig, 2007, ISBN 978-3-446-40971-2
Bemerkungen:	---

Mobile Robotik

Kürzel:	MRO
Untertitel:	Grundlegende Konzepte und Verfahren zur Entwicklung von Applikationen für mobile Roboter
Studiensemester:	5. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Hartmut Surmann
Dozent(in):	Prof. Dr. Hartmut Surmann
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: 5; MI: -; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	keine
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung, Grundlagen der Mathematik für Informatiker, Lineare Algebra
Angestrebte Lernergebnisse:	Die Studierenden kennen die Begriffe und Komponenten von mobilen Robotern sowie die Konzepte und Methoden der Programmierung und können diese effektiv und strukturiert bei der Entwicklung eigener Steuerungsprogramme einsetzen. Sie wissen wie unterschiedliche Sensordaten fusioniert werden und mobile Systeme navigieren und sich selbst lokalisieren. Sie kennen die Gefahren beim Umgang mit mobilen Systemen und die Wichtigkeit der Einhaltung von Vorschriften sowohl auf technischer als auch sozialer Ebene. Die Studierenden sind zudem in der Lage, sich selbstständig und zügig in unterschiedliche Arten von mobilen Robotern und deren Programmierumgebung einzuarbeiten. Betriebssystem: Linux + ROS, Lehrsprache ist C / C++, Python.

Inhalt:	Einführung / Begriffsbildung Mobile Roboter • Sensorik • Aktuatorik • Lokalisierung • Kartierung • Navigation • Planung
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• J. Hertzberg, K. Lingemann, A. Nüchter: „Mobile Roboter: Eine Einführung aus Sicht der Informatik“, ISBN 978-3642017254 • B. Siciliano, O. Khatib (Eds.): „Handbook of Robotic“, ISBN 978-3-540-23957-4 • Craig, J.J. (2004), „Introduction to Robotics: Mechanics and Control (3rd Edition)“, 8, 2004. Prentice Hall • R. Siegwart „Introduction to Autonomous Mobile Robots“, MIT Press, ISBN: 978-0-262-19502 -7 • S. Thrun, W. Burgard, D. Fox: „Probabilistic Robotics“, ISBN 978-0262201629
Bemerkungen:	---

Modellbasierter Entwurf von Regelsystemen

Kürzel:	MBE
Untertitel:	---
Studiensemester:	5., ggf. 4. (Wahlpflicht im Bachelor)
Modulverantwortliche(r):	Prof. Dr. Ekkehard Schrey
Dozent(in):	Prof. Dr. Ekkehard Schrey
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: - ; I/TI: WP ; MI: - ; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 58 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 122 Zeitstunden
Leistungspunkte:	6
Turnus:	In der Regel jährlich im Wintersemester
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Grundlagen der Mathematik für Informatiker, Mathematik für Informatiker, Technische Grundlagen der Informatik, Algorithmen und Datenstrukturen, Einführung in die Programmierung, Einführung in die Systemtheorie
Angestrebte Lernergebnisse:	Die Studierenden erlangen Grundkenntnisse der Beschreibung linearer und nichtlinearer Systeme und sind in der Lage, mathematische Modelle einfacher physikalischer Systeme aufzustellen erlangen Kenntnisse über die grundlegenden Verfahren der numerischen Simulation lernen ein Werkzeug zur Simulation technischer Systeme kennen und sind in der Lage, das Werkzeug sachgerecht einzusetzen lernen ein Werkzeug zur automatischen Codegenerierung für Regelsysteme kennen und sind in der Lage, dieses Werkzeug sachgerecht einzusetzen

	erlangen Kenntnisse über mögliche Beschreibungsformen digitaler Regelungen und sind in der Lage, diese Kenntnisse bei der Entwicklung eines Regelsystems für eine einfaches Umgebungssystem sachgerecht einzusetzen
Inhalt:	• Einsatzgebiet technischer Simulation • Beschreibungsformen zeitkontinuierlicher technischer Systeme (lineare und nichtlineare Systeme) • Werkzeuge zur Simulation zeitkontinuierlicher Systeme • Überblick über numerische Verfahren im Rahmen der Simulation (Verfahren zur numerischen Lösung von Differentialgleichungen, Schrittweitensteuerung, Verfahren zur Nullstellensuche) • Überblick über Systeme zur automatischen Codegenerierung • Beschreibungsformen zeitdiskreter Regelsysteme • Modellbasierter Entwurf zeitdiskreter Systeme
Studien- / Prüfungsleistungen:	• Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: • Vorrechnung einer gewissen Anzahl von Übungsaufgaben. Die Anzahl der Aufgaben wird zu Beginn der Veranstaltung so festgelegt, dass jeder Studierende diese Anzahl gut bewältigen kann. • Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. • Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine • Prüfungsleistungen: Klausur (90 Min.), ggf. mündliche Prüfung
Literatur:	Kramer, Neculau; Simulationstechnik; Hanser-Verlag; ISBN 3- 446-19235-2 Gipser; Systemdynamik und Simulation; Teubner-Verlag; ISBN 3-519-02743-7 Preuß, Wenisch; Numerische Mathematik; Fachbuchverlag Leipzig 2001; ISBN 3-446-21375-9 Kahlert, J.; Simulation technischer Systeme; Vieweg 2004; ISBN 3-528-03964-7

	Berns, Schürmann, Trapp; Eingebettete Systeme; Vieweg& Teubner 2010; ISBN 978-3-8348-0422-8
Bemerkungen:	---

Objektorientierte Programmierung mit C++

Kürzel:	OPC
Untertitel:	---
Studiensemester:	5. (Wahlpflicht im Bachelor)
Modulverantwortliche(r):	Studiengangsbeauftragte/r Informatik
Dozent(in):	Prof. Dr. Reinhard Wierich oder Lehrbeauftragte/r
Sprache:	Deutsch; bei englischsprachigen Teilnehmern Englisch
Zuordnung zum Curriculum:	I/PI: WP; I/TI: WP; MI: WP; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Standard
Arbeitsaufwand:	Standard
Leistungspunkte:	6
Turnus:	Wintersemester, unregelmäßig
Teilnehmerzahl:	Standard
Anmeldungsmodalitäten:	Siehe Aushang
Voraussetzungen nach Prüfungsordnung:	Erfolgreich absolviertes Praktikum
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung
Angestrebte Lernergebnisse:	• Verständnis für die Konzepte der Objektorientierten Programmierung in C++ • Sicherer Umgang mit der Syntax von C++, speziell Pointern und Referenzen • Verständnis des Operator Overloading – Konzepts, speziell in Verbindung mit Containern • Eingeübter Umgang mit Templates und entsprechenden Bibliotheken
Inhalt:	• Syntaxerweiterungen gegenüber C • Konzeptuelle Unterschiede gegenüber JAVA • Default- und Kopier-Konstruktor, Initialisierungsliste • Operator Overloading • Vererbung und Mehrfachvererbung • Templates, Container und Iteratoren • Arbeiten mit der Standard Template Library • Fortgeschrittene Template-Anwendungen • Exception Handling in C++
Studien- / Prüfungsleistungen:	Studienleistungen: Erfolgreich absolviertes Praktikum Prüfungsleistungen: Klausur (120 Min.)

Literatur:	Prata, Stephen: C++**.** te-wi Verlag, ISBN 3-89362-701-4. Prinz, P. *et al*.,: Objektorientiert programmieren mit ANSI C++. Verlag Prentice Hall, 1998, ISBN 3-8272-9560-2. Stroustrups, Die C++ Programmiersprache. Addison-Wesley, 2000. Koenig *et al*., Intensivkurs C++. Addison-Wesley, 2003, ISBN 3-8273-7029-9
Bemerkungen:	---

Parallele Programmierung

Kürzel:	PAP			
Untertitel:	Grundlegende Methoden und Algorithmen paralleler Programmierung			
Studiensemester:	4. (Wahlpflicht im Bachelor)			
Modulverantwortliche(r):	Prof. Dr. Hartmut Surmann			
Dozent(in):	Prof. Dr. Hartmut Surmann			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	WP	WP	WP	-
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum			
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden			
Leistungspunkte:	6			
Turnus:	Sommersemester, jährlich			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	Keine			
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen			
Empfohlene Voraussetzungen (Modulprüfungen):	Algorithmen und Datenstrukturen, Einführung in die Programmierung, Objektorientierte Programmierung, Prozedurale Programmierung, Grundlagen der Mathematik für Informatiker			
Angestrebte Lernergebnisse:	Die Studierenden lernen die Grundlagen der Parallelen Programmierung kennen und können parallele Programme entwickeln und testen. Weiterhin lernen sie klassische Algorithmen zu parallelisieren und innerhalb eines Rechners und über mehrere Rechner hinweg parallel zu verteilen. Neben der Programmiermethodik und dem Design lernen die Studierenden die speziellen Probleme und Fragestellungen bei der parallelen Programmierung kennen, insbesondere die schwierigere Fehleranalyse.			
Inhalt:	Mit dem Aufkommen von leistungsfähigen Grafikkarten, Multicore- Prozessoren in PCs und Smartphones erhält die Parallel Programmierung einen immer höheren Stellenwert und ist für eine Vielzahl von Anwendungen			

	interessant. Die Vorlesung deckt die nachfolgenden Themengebieten ab: • Grundlagen paralleler Programmierung • Parallele Architekturen • Design paralleler Algorithmen • Threads • OpenMP • MPI • OpenCL • CUDA
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• Thomas Rauber: “Parallele Programmierung”, Springer Verlag, ISBN 978-3-540-46549-2. • D. E. Culler, J. Pal Singh, A. Gupta: “Parallel Computer Architecture: A Hardware/Software Approach”, Morgan Kaufmann, ISBN 1-55860-343-3. • I. Foster. ”Designing and Building Parallel Programs”, Addison Wesley, ISBN 978-0-201-57594-1. • P. Pacheco: “An Introduction to Parallel Programming”, Morgan Kaufmann, ISBN 978-0-123-74260-5 • Patterson, Hennessy, Bode: “Rechnerorganisation und –entwurf”, Standardwerk, Spektrum Akademischer Verlag, ISBN 978-3-827-41595-0 .
Bemerkungen:	---

Practical Security Attacks and Exploitation

Kürzel:	PRAX			
Untertitel:	Practical Security Attacks and Exploitation			
Studiensemester:	4. oder 5. (Wahlpflicht im Bachelor)			
Modulverantwortliche(r):	Prof. Dr. Christian Dietrich			
Dozent(in):	Prof. Dr. Christian Dietrich			
Sprache:	Englisch oder deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	WP	WP	WP	WP
Lehrform / SWS:	1 SWS Vorlesung, 3 SWS Praktikum			
Gruppengröße:	Vorlesung: nicht begrenzt, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 52 h Eigenstudium inkl. Projekt/Ausarbeitung und Prüfungsvorbereitung: 128 h			
Leistungspunkte:	6			
Turnus:	Sommer- und Wintersemester, unregelmäßig			
Teilnehmerzahl:	Keine Beschränkung			
Anmeldungsmodalitäten:	Anmeldung via Moodle			
Voraussetzungen nach Prüfungsordnung:	Keine besonderen			
Empfohlene Voraussetzungen (Modulprüfungen):	Grundlegende IT-Sicherheitskenntnisse			
Angestrebte Lernergebnisse:	• Verständnis gängiger Verfahren zur Systemsicherheit, Systemintegrität und zum Softwareschutz • Anwenden von Mechanismen zur Identifikation und Ausnutzung von Software-Schwachstellen • Anwenden von Angriffstechniken in Computernetzwerken • Erlangen von Kenntnissen im Bereich der Schadsoftware-Erkennung und -Abwehr • Teilnahme an einem Capture-the-Flag-Wettbewerb (z.B. Cyber Security Challenge)			
Inhalt:	Die Studierenden lernen die Anwendbarkeit und Grenzen von sicherheitsrelevanten Angriffen gegen Systeme, Netzwerkprotokolle und Software. Dabei werden die folgenden Themen behandelt: • Introduction to vulnerability research • Reconnaissance and scanning • Infection vectors			

	• System security and operational security • Software security • Network security • Denial-of-Service attacks • Web security • Incident response Lerneinheiten bestehen jeweils aus einer Einführung in Form mindestens einer Vorlesungseinheit sowie Aufgaben, die im Praktikum gelöst werden müssen. Darüber hinaus müssen die Studierenden selbst verwundbare Beispiele als Aufgaben entwerfen, die beispielsweise im Rahmen eines eigenen CTF-Wettbewerbs eingesetzt werden könnten.
Studien- / Prüfungsleistungen:	Studienleistungen: Die Studierenden können während des Praktikums Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der Vorlesung bekannt gegeben. Prüfungsleistungen: Ausarbeitung sowie Klausur oder mündliche Prüfung
Literatur:	Eckert, C.: *IT-Sicherheit. Konzepte, Verfahren, Protokolle.* Oldenbourg, München, aktuellste Auflage Erickson, J.: *Hacking - The Art of Exploitation*. No Starch Press; aktuellste Auflage Aktuelle wissenschaftliche Publikationen
Bemerkungen:	---

Systemtheorie

Kürzel:	SYT			
Untertitel:	---			
Studiensemester:	4., ggf. 5. (Wahlpflicht im Bachelor)			
Modulverantwortliche(r):	Prof. Dr. Ekkehard Schrey			
Dozent(in):	Prof. Dr. Ekkehard Schrey			
Sprache:	Deutsch			
Zuordnung zum Curriculum:	I/PI	I/TI	MI	WI
	-	WP	-	-
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum			
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20			
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 67 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 113 Zeitstunden			
Leistungspunkte:	6			
Turnus:	In der Regel jährlich im Sommersemester			
Teilnehmerzahl:	Nicht begrenzt			
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul			
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen			
Empfohlene Voraussetzungen (Modulprüfungen):	Grundlagen der Mathematik für Informatiker, Mathematik für Medieninformatiker			
Angestrebte Lernergebnisse:	Die Studierenden erlangen Kenntnisse über die Grundlagen, Methodik und Hilfsmittel bei der Beschreibung linearer, zeitinvarianter technischer Systeme mit Hilfe der Werkzeuge der Systemtheorie. Sie sind in der Lage, die erlernten Verfahren selbständig bei der Beschreibung und Analyse von linearen Systemen anzuwenden.			

Inhalt:	• Systembeschreibung mit linearen Differentialgleichungen • Aufstellen von Wirkungsplänen • Faltungsintegral und Übertragungsfunktion • Fourier-Transformation, Laplace-Transformation • Ideale Übertragungssysteme und ihre Eigenschaften • Zeitdiskrete Systeme • z-Transformation • Zeitdiskrete Übertragungsfunktionen
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: Vorrechnung einer gewissen Anzahl von Übungsaufgaben. Die Anzahl der Aufgaben wird zu Beginn der Veranstaltung so festgelegt, dass jeder Studierende diese Anzahl gut bewältigen kann. Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.)
Literatur:	Rennert, I, Bundschuh, B; Signale und Systeme, Fachbuchverlag Leipzig, 2013, ISBN 978-3-446-43327-4 Lunze, J; Regelungstechnik 1, Springer 2007, ISBN 978-3-540-68907-2 Reuter, M, Zacher, S.; Regelungstechnik für Ingenieure, ViewegTeubner, 2008, ISBN 978-3-8348-0018-3 Weitere Literatur wird noch bekannt gegeben
Bemerkungen:	---

Zeitdiskrete Regelsysteme

Kürzel:	ZDR
Untertitel:	---
Studiensemester:	5., ggf. 4. (Wahlpflicht im Bachelor)
Modulverantwortliche(r):	Prof. Dr. Ekkehard Schrey
Dozent(in):	Prof. Dr. Ekkehard Schrey
Sprache:	Deutsch
Zuordnung zum Curriculum:	I/PI: - ; I/TI: WP ; MI: - ; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Präsenzstudium inkl. Modulprüfung: 58 Zeitstunden Eigenstudium inkl. Prüfungsvorbereitung: 122 Zeitstunden
Leistungspunkte:	6
Turnus:	In der Regel jährlich im Wintersemester
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Grundlagen der Mathematik für Informatiker, Mathematik für Informatiker, Technische Grundlagen der Informatik, Algorithmen und Datenstrukturen, Einführung in die Programmierung, Einführung in die Systemtheorie
Angestrebte Lernergebnisse:	Die Studierenden • erlangen Kenntnisse der Beschreibung linearer Systeme mit zeitdiskreten Funktionen und sind in der Lage, Systeme in einer zeitdiskreten Form (Differenzengleichungen, zeitdiskrete Übertragungsfunktionen) zu beschreiben • erlangen Kenntnisse über die Eigenschaftsmerkmale zeitdiskreter Systeme und sind in der Lage, die Merkmale von gegeben Systemen selbständig zu erkennen • erlangen Kenntnisse über grundlegende Regelalgorithmen für zeitdiskrete Systeme sowie deren Kenngrößen und sind in der Lage,

	die Kenntnisse bei der Auslegung eines Reglers selbständig anzuwenden • erlangen Kenntnisse, wie aus zeitdiskreten Regelalgorithmen systematisch Software für ein Zielsystem generiert werden kann und sind in der Lage, diese bei der Programmierung eines Systems selbständig anzuwenden
Inhalt:	• Einführung in die kontinuierliche Regelungstechnik • Eigenschaften und Qualitätsmerkmale von Regelkreisen • Beschreibung von Systemen mit Differenzengleichungen • Kurzeinführung in die z-Transformation • Differenzengleichungen und z-Übertragungsfunktionen • Zustandsdarstellung von Systemen • Regelungen für Systeme mit einem Ein- und Ausgang (Parameteroptimierte Regler, Strukturoptimale Regler) • Überblick Verfahren Mehrgrößenregelungen • Praktische Probleme bei der Reglerprogrammierung • Empfindlichkeit und Robustheit von Reglern
Studien- / Prüfungsleistungen:	Studierende erhalten für die folgenden freiwillig zu erbringenden semesterbegleitenden Leistungen ein modulspezifisches Teilnahmezertifikat: • Vorrechnung einer gewissen Anzahl von Übungsaufgaben. Die Anzahl der Aufgaben wird zu Beginn der Veranstaltung so festgelegt, dass jeder Studierende diese Anzahl gut bewältigen kann. • Absolvierung des Praktikums. Eine zu Beginn der Veranstaltung festgelegte Anzahl von Praktikumsaufgaben muss ausprogrammiert und vorgeführt werden. Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.); alternativ mündliche Prüfung (30 Min.)
Literatur:	Isermann, R.; Digitale Regelsysteme, Springer-Verlag 1988, ISBN 3-540-16596-7 Vogel, P.; Systemtheorie ohne Ballast – Zeitdiskrete LTI-Systeme; Springer-Verlag 2011, ISBN 978-3-642-16045-5 Roderer, H., Pecher, A.; Digitale Signalverarbeitung, Vogel Buchverlag 2010, ISBN 978-3-8343-3115-1

Bemerkungen:	---

Wahlpflichtkatalog Informatik

Betrieb komplexer verteilter Systeme

Kürzel:	BKV
Untertitel:	---
Studiensemester:	4 (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Andreas Cramer
Dozent(in):	Prof. Dr. Andreas Cramer
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 2 SWS Praktikum
Gruppengröße:	Vorlesung nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 56 Zeitstunden Selbststudium: 124 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Betriebssysteme, Rechnernetze
Angestrebte Lernergebnisse:	Die Studierenden lernen unterschiedliche Technologien und Konzepte kennen, die für den Betrieb großer IT-Infrastrukturen notwendig sind und bekommen erste praktische Erfahrungen mit deren Anwendung. Sie erlangen die Fähigkeit, neue Konzepte im Umfeld des IT-Betriebs schnell begreifen, einordnen und bewerten zu können.
Inhalt:	Einführung Speichernetze Virtualisierung System-Management
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: keine Prüfungsleistungen: Klausur oder mündliche Prüfung
Literatur:	Bekanntgabe in der Vorlesung
Bemerkungen:	---

Einführung in die Bildverarbeitung

Kürzel:	BV		
Untertitel:	Grundlegende Verfahren und Algorithmen der digitalen Bildverarbeitung		
Studiensemester:	5. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr.-Ing. Dipl. Inform. Hartmut Surmann		
Dozent(in):	Prof. Dr.-Ing. Dipl. Inform. Hartmut Surmann		
Sprache:	deutsch, englisch bei Bedarf		
Zuordnung zum Curriculum:	IN	ID	WI
	WP	-	WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum		
Gruppengröße:	Standard		
Arbeitsaufwand:	Kontaktzeit: 75 Zeitstunden Selbststudium: 105 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Wintersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Keine		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Objektorientierte Programmierung, Algorithmen und Datenstrukturen, Grundlagen der Mathematik für Informatiker, Lineare Algebra		
Angestrebte Lernergebnisse:	Die Studierenden lernen die Begriffe und Verfahren der digitalen Bildverarbeitung und die Konzepte und Methoden deren Programmierung kennen. Sie können diese effektiv und strukturiert bei der Entwicklung eigener Bildverarbeitungsprogramme einsetzen. Neben der Programmiermethodik lernen die Studierenden die Verwendung von Bibliotheken (OpenCV, CNN‘s) kennen und können diese für die Entwicklung eigener Lösungen einsetzten.		
Inhalt:	• Grundlagen / Begriffsbildung • Kameras • Bildverarbeitungsoperationen • Bildsegmentierung • Merkmale von Objekten • Klassifikation		

	• Neuronale Netze, CNNs • Lehrsprachen: C / C++, Python, ipython notebooks
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• J. Steinmüller: „Bildanalyse“, Springer Verlag, ISBN 978-3540797425.- A. Nischwitz, P. Haberäcker: „Computergrafik und Bildverarbeitung, Band II Bildverarbeitung“, TeubnerVerlag, ISBN 978-3-834-81712-9 • A. Kaehler, G Bradski: "Learning OpenCV 3: Computer Vision in C++ with the OpenCV", 978-1491937990
Bemerkungen:	https://moodle.w-hs.de/course/index.php?categoryid=232

Data on the Web

Kürzel:	DOW
Untertitel:	-
Studiensemester:	4. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Katja Zeume
Dozent(in):	Prof. Dr. Katja Zeume
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 56 Zeitstunden Selbststudium: 124 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester (jährlich)
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Über den dazugehörenden Moodle-Kurs
Voraussetzungen nach Prüfungsordnung:	keine
Empfohlene Voraussetzungen (Modulprüfungen):	Datenbanksysteme
Angestrebte Lernergebnisse:	Die Studierenden kennen die gängigen Anwendungsfelder der webbasierten Datenverarbeitung und die spezifischen Probleme die im Einsatz auftauchen. Dabei lernen die Studierenden ihre Kenntnisse über relationale Datenbanksysteme mit weiterführenden Technologien zu erweitern und auf nicht-relationale Anwendungen zu übertragen.
Inhalt:	Die Veranstaltung bietet eine Vertiefung in aktuelle erweiterte Datenbankformate und Anfragesprachen im Kontext von webbasierten und Cloud-Anwendungen. • Bedeutung semi-strukturierte Formate (am Beispiel XML oder JSON), sowie passenden Anfrage- und Schemasprachen. • (Wahlweise) Einführung unstrukturierte Datenbankformate (sog. NOSQL Datenbanken) am Beispiel Graphdatenbanken. • (Wahlweise) Überblick über weitere Formate aus dem Bereich TextDB und StreamingDB.

	• (Wahlweise) Aspekte von Cloud Maturity für Datenbasierte Anwendungen im Web Die einzelnen Themen werden mit Anwendungsfällen aus der Praxis motiviert und auf ihre Vor- und Nachteile verglichen.
Studien- / Prüfungsleistungen:	Klausur (75min)
Literatur:	• Leskovec, Rajaraman, Ullman. Mining of Massive Datasets • Foundations of Databases, Serge Abiteboul, Rick Hull, Victor Vianu, 1995.
Bemerkungen:	-

Data Science in Practice

Kürzel:	DSP
Untertitel:	
Studiensemester:	4. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Laura Anderle
Dozent(in):	Prof. Dr. Laura Anderle
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: -
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Praktikum
Gruppengröße:	20 Personen
Arbeitsaufwand:	Kontaktzeit: 42 Zeitstunden Selbststudium: 138 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Erscheinen zur ersten Vorlesung und Anmeldung zum Kurs-Moodleraum
Voraussetzungen nach Prüfungsordnung:	keine
Empfohlene Voraussetzungen (Modulprüfungen):	MGI, SLA
Angestrebte Lernergebnisse:	Die Studierenden sind nach Absolvieren des Moduls in der Lage, eine anwendungsorientierte Fragestellung mit Hilfe von Data Science-Methoden zu beantworten. Sie verstehen die angewandten Methoden und deren Einsetzbarkeit in der Praxis und können Daten, Methoden und Ergebnisse fachfremden erläutern und diskutieren. Fragestellungen und Datensätze entstammen in der Regel einem gesundheits-, gesellschafts- oder ingenieurwissenschaftlichen Kontext. Insbesondere sind die Studierenden in der Lage, • Daten, Methoden und Ergebnisse mit Nicht-Informatikern zu diskutieren, • Daten zu bereinigen und für Analysen vorzubereiten, • explorative und deskriptive Datenanalysen durchzuführen zu entscheiden, ob eine gegebene Fragestellung am besten mit Regressions-, Klassifikations- oder

	Clusteringmethoden beantwortet werden kann, diese anzuwenden und die Ergebnisse zu interpretieren.
Inhalt:	• Vorstellung des behandelten Datensatzes und seiner fachlichen Hintergründe • Datenexploration, -visualisierung und deskriptive Analysen • Überblick über zur Verfügung stehende Methodiken • Formulierung geeigneter Fragestellungen und Methodenauswahl • Einarbeitung in die ausgewählten Methoden • Datenanalyse • Vorstellung und Diskussion der Ergebnisse Programmieranteile erfolgen in Python oder R. Die konkrete Methodenauswahl erfolgt im Kurs im Dialog mit den Studierenden. Je nach Fragestellung wird angestrebt, Studierende, Praktiker*innen oder Wissenschaftler*innen anderer Fachrichtungen zu einzelnen Kursterminen hinzuzuziehen.
Studien- / Prüfungsleistungen:	Klausur und/oder schriftliche Ausarbeitung und/oder mündliche Prüfung
Literatur:	• Benjamin M. Abdel-Karim: Data Science - Best Practices mit Python (Springer 2022) • Manas A. Pathak: Beginning Data Science with R (Springer 2014) • Joel Grus: Einführung in Data Science – Grundprinzipien der Datenanalyse mit Python (O'Reilly 2019)
Bemerkungen:	---

Einführung in die medizinische Informatik

Kürzel:	EMI
Untertitel:	
Studiensemester:	4.(Bachelor)
Modulverantwortliche(r):	Prof. Dr. Laura Anderle
Dozent(in):	Prof. Dr. Laura Anderle
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: -
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Praktikum
Gruppengröße:	Nicht begrenzt
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Erscheinen zur ersten Vorlesung und Anmeldung zum Kurs-Moodleraum
Voraussetzungen nach Prüfungsordnung:	keine
Empfohlene Voraussetzungen (Modulprüfungen):	MGI, SLA
Angestrebte Lernergebnisse:	Die Studierenden sind nach Absolvieren des Moduls in der Lage, das anwendungsbezogene Fachgebiet "Medizinische Informatik" zu überblicken. Sie können Zusammenhänge zwischen den verschiedenen Anwendungsbereichen und Teilgebieten herstellen und kennen die notwendigen IT-Grundlagen der medizinischen Informatik. Insbesondere sind die Studierenden in der Lage • Ziele, Nutzen und Aspekte von medizinischen IT-Anwendungen zu erklären, • med. Vorgänge/Prozesse zu modellieren und die Bedeutung von Prozessunterstützung durch IT zu erklären • zu erklären, welche medizinischen bildgebenden Verfahren und Biosignale es gibt und welche mathematischen Operationen bei deren Übernahme in Rechnersysteme nötig sind,

	• Ziele und Vorgehen bei klinischen und epidemiologischen Studien zu beschreiben und den IT-Einsatz hierzu darzustellen • Einfache Programme zu Teilaspekten der medizinischen Informatik zu schreiben und einen ersten Überblick über rechtliche Aspekte medizinischer Software zu geben.
Inhalt:	• Überblick über die Teilgebiete der medizinischen Informatik • Medizinische Prozesse, Dokumentation und Informationssysteme • Bildgebende Verfahren und Biosignale • Klinische und epidemiologische Studien Einführung in rechtliche Aspekte
Studien- / Prüfungsleistungen:	Klausur und/oder schriftliche Ausarbeitung und/oder mündliche Prüfung
Literatur:	• Lehmann et al.: Handbuch der Medizinischen Informatik, (Hanser 2005) • Martin Dugas und Katrin Schmidt: Medizinische Informatik und Bioinformatik (Springer 2002)
Bemerkungen:	---

Einführung in die Robotik

Kürzel:	ERO		
Untertitel:	Grundlegende Konzepte und Verfahren zur Entwicklung von Applikationen für Roboter / Manipulatoren		
Studiensemester:	4. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr.-Ing. Dipl. Inform. Hartmut Surmann		
Dozent(in):	Prof. Dr.-Ing. Dipl. Inform. Hartmut Surmann		
Sprache:	Deutsch, englisch bei Bedarf		
Zuordnung zum Curriculum:	IN	ID	WI
	WP	-	WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum		
Gruppengröße:	Standard		
Arbeitsaufwand:	Kontaktzeit: 70 Zeitstunden Selbststudium: 110 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Sommersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Keine		
Voraussetzungen nach Prüfungsordnung:	Keine		
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Objektorientierte Programmierung, Algorithmen und Datenstrukturen, Mathematische Grundlagen, Statistik und Lineare Algebra		
Angestrebte Lernergebnisse:	Die Studierenden lernen die Grundlagen, Komponenten und Begriffe von Industrierobotern und kollaborativen Robotern kennen. Sie lernen Konzepte und Methoden der Programmierung und können diese effektiv und strukturiert bei der Entwicklung eigener Steuerungsprogramme einsetzten. Sie kennen die Gefahren und Herausforderungen beim Einsatz von Industrierobotern und verstehen die Wichtigkeit der Einhaltung von Vorschriften. Neben der Programmiermethodik lernen die Studierenden die Verwendung von Bibliotheken des Roboter Frameworks ROS (Robot Operation System) kennen.		
Inhalt:	• Grundlagen der Industrierobotik / Manipulatortechnik, kollaborativer Roboter • Begriffsbildung und Komponenten		

	• Beschreibung einer Roboterstellung • Transformation zwischen Roboter- und Weltkoordinaten, • Kinematic, inverse Kinematic • Roboterprogrammierung, • Roboterframework ROS, • Bewegungsart und Interpolation • Betriebssytem: Linux + ROS; Lehrsprachen sind C / C++, Python, ipython notebooks
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• Wolfgang Weber: Industrieroboter, Methoden der Steuerung und Regelung, Hanser Verlag, 4. Auflage, ISBN 978-3-446-41031-2 • Quigley, M: Programming Robots with ROS: A Practical Introduction to the Robot Operating System • P.I. Corke, “Robotics, Vision & Control”, Springer 2017, ISBN 978-3-319-54413-7 und Robotics Tollbox for Python- Introduction to Robotics: Mechanics and Control: Global Edition, 3rd Edition • Bruno Siciliano, Oussama Khatib (Eds.): Handbook of Robotic, ISBN 978-3-540-23957-4
Bemerkungen:	https://moodle.w-hs.de/course/index.php?categoryid=232

Internet-Protokolle

Kürzel:	INP		
Untertitel:	Grundlegende Kenntnisse über die Aufgaben, Prinzipien, Mechanismen und Architekturen auf den unterschiedlichen Kommunikationsebenen und Grundlagen von Verteilten-Systemen		
Studiensemester:	5. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr. (TU NN) Norbert Pohlmann		
Dozent(in):	Prof. Dr. (TU NN) Norbert Pohlmann		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	WP	-	WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum		
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 30, Praktikum: 20		
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Wintersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	Keine		
Angestrebte Lernergebnisse:	• Gutes Verständnis für die fundamentalen Kommunikationsarchitekturen und -protokolle des Internets. • Erlangen von Kenntnissen über die Aufgaben, Prinzipien, Mechanismen und Architekturen auf den unterschiedlichen Kommunikationsebenen. • Gewinnen von praktischen Erfahrungen über die Kommunikationsprotokolle, Kommunikationsdienste und -anwendungen durch Versuche und mit Hilfe von Protokollanalysen. • Erleben der Notwendigkeit und Wichtigkeit der Lehrinhalte.		

Inhalt:	• Einführung: Begriffe, geschichtliche Entwicklung, Beispiele für Netzwerke, die Zukunft von Netzwerken und des Internets • Das ISO- und TCP/IP-Referenzmodell: Instanzen, Dienste, Protokolle, Paketstrukturen; Schichtenaufgaben • Netzkoppelelemente: Repeater, Hubs, Bridges, Switches, Router, Gateway • Vermittlungsebene: Aufgaben der Vermittlungsebene (IP, ARP, ICMP, Routingprotokolle); Begriffe/Mechanismen der Vermittlungstechnik (Warteschlangen, Routingverfahren, Traffic Shaping, Scheduling, Call admission control); Quality of Service in IP-Netzen (Idee, Konzept, IntServ, RSVP, DiffServ, MLPS) • Transportebene: Dienste und Mechanismen der Transportschicht (TCP, UDP; RTP); Sequenz- und Bestätigungsnummern, Prüfsumme, Zeitüberwachung, Segmentierung, Stream-Service, Sliding-Windows-Technik, Slow-Start, Congestion Windows, Delayed acknowledgement, Nagle Algorithmus • Anwendungsebene: DNS (Domain Name Service), SMTP (E- Mail), HTTP (World Wide Web), SIP (Session Initiation Protocol) Pro Anwendungsdienst: Kommandos, Nachrichten/Datentypen, Verbindungen/Kommunikation, Besonderheiten; Protokollanalysen und deren Bewertung • Client-Server- und P2P-Architektur • Struktur und Aufbau des Internets (AS, Arten von ASe, Verbindungen, CDN, …) • Grundlagen von Verteilten Systemen (Motivation, Ziele, Konzepte, Beispiele, …)
Studien- / Prüfungsleistungen:	Studienleistungen: Erfolgreich absolviertes Praktikum als Vorleistung für die Prüfungszulassung Prüfungsleistungen: Klausur (90 Min.)
Literatur:	• Tanenbaum, A.: "Computernetzwerke"; Prentice Hall, 2003; ISBN: 3- 8273-7046-9 • Tanenbaum, A.; van Stehen, M.: "Verteilte Systeme - Grundlagen und Paradigmen"; Prentice Hall, 2003; ISBN: 3-8273-7057-4 • Proebster, W: "Rechnernetze - Technik, Protokolle, Systeme, Anwendungen"; Oldenbourg Verlag; ISBN: 3-486-25777-3 • Müller, G.; Eymann, T.; Kreutzer, M.: "Telematik- und Kommunikationssysteme in der vernetzten

	Wirtschaft"; Oldenbourg Verlag; ISBN: 3-486-25888-5 • Wander, K.; "Protokolle und Dienste der Informationstechnologie"; Interest Verlag; ISBN: 3-8245-0412-X • S. Feld, N. Pohlmann, M. Sparenberg, B. Wichmann: „Analyzing G-20´Key Autonomous Systems and their Intermeshing using AS-Analyzer". In Proceedings of the ISSE 2012 - Securing Electronic Business Processes - Highlights of the Information Security Solutions Europe 2012 Conference, Eds.: N. Pohlmann, H. Reimer, W. Schneider; Springer Vieweg Verlag, Wiesbaden 2012
Bemerkungen:	-

IT-Recht

Kürzel:	ITR		
Untertitel:			
Studiensemester:	5 (Bachelor)		
Modulverantwortliche(r):	Studiengangsbeauftragte/r Wirtschaftsinformatik		
Dozent(in):	Lehrbeauftragte/r		
Sprache:	deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	WP	-	5
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung		
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 40		
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Wintersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul		
Voraussetzungen nach Prüfungsordnung:	keine		
Empfohlene Voraussetzungen (Modulprüfungen):	keine		
Angestrebte Lernergebnisse:	Die Studierenden werden in die Lage versetzt: die relevanten rechtlichen Aspekte und gesetzlichen Regelungen als Randbedingung in ihre berufliche Arbeit einbeziehen können, zu wissen, welche datenschutzrechtlichen Vorgaben es bei der Speicherung personenbezogener Daten gibt oder welche rechtlichen Regeln bei der Gestaltung und Programmierung von Internet-Auftritten einzuhalten sind.		
Inhalt:	Rechtliche Aspekte bei der Erstellung und Anwendung von Softwareprodukten aller Art, Internet-, Datenschutz- und Urheberrecht, die für die behandelten Rechtsfelder maßgeblichen europäischen und deutschen Gesetze.		
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.) Prüfungsleistungen: Klausur (90 Minuten)		

Literatur:	Nach Bekanntgabe in der Vorlesung.
Bemerkungen:	—-

Grundlagen der IT-Sicherheit

Kürzel:	ITS
Untertitel:	Grundlegende Konzepte, Architekturen, Prinzipien und Funktionsweisen von Sicherheitskomponenten und -systemen
Studiensemester:	5. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. (TU NN) Norbert Pohlmann
Dozent(in):	Prof. Dr. (TU NN) Norbert Pohlmann
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Übung: 40, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden
Leistungspunkte:	6
Turnus:	Wintersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen
Empfohlene Voraussetzungen (Modulprüfungen):	Keine
Angestrebte Lernergebnisse:	• Gutes Verständnis von möglichen Angriffen und geeigneten Gegenmaßnahmen in der IT • Erlangen von Kenntnissen über den Aufbau, die Prinzipien, die Architektur und die Funktionsweise von Sicherheitskomponenten und -systemen • Sammeln von Erfahrungen bei der Ausarbeitung und Präsentation von neuen Themen aus dem Bereich IT-Sicherheit • Gewinnen von praktischen Erfahrungen über die Nutzung und die Wirkung von Sicherheitssystemen • Erleben der Notwendigkeit und Wichtigkeit der IT-Sicherheit
Inhalt:	• Einführung: IT-Sicherheitslage, Cyber-Sicherheitsstrategien, Cyber-Sicherheitsbedürfnisse, Angreifer – Motivationen, Kategorien und

	Angriffsvektoren, Pareto-Prinzip: Cyber-Sicherheit, Cyber-Sicherheitsschäden • Kryptographie und technologische Grundlagen für Schutzmaßnahmen: Private-Key-Verfahren, Public-Key-Verfahren, Kryptoanalyse, Hashfunktionen, Schlüsselgenerierung • Sicherheitsmodule (SmartCards, TPM, high-security und high-performence Lösungen) • Identifikations- und Authentikationsverfahren: Grundsätzliche Prinzipien sowie unterschiedliche Algorithmen und Verfahren • ID-Management (Idee, Ziel, Konzepte) • ID-Cards (Neuer Personalausweis, Smart-eID …) • Self-Sovereign Identity (SSI)
Studien- / Prüfungsleistungen:	Studienleistungen: Erfolgreich absolviertes Praktikum als Vorleistung für die Prüfungszulassung Prüfungsleistungen: Klausur (90 Min.)
Literatur:	• N. Pohlmann: „Cyber-Sicherheit - Das Lehrbuch für Konzepte, Mechanismen, Architekturen und Eigenschaften von Cyber-Sicherheitssystemen in der Digitalisierung“ 2. Auflage, Springer Vieweg Verlag, Wiesbaden 2022 • Pohlmann, N.: Firewall-Systeme - Sicherheit für Internet und Intranet, E- Mail-Security, Virtual Private Network, Intrution Detection-System, Personal Firewalls. 5. aktualisierte und erweiterte Auflage; ISBN 3- 8266-0988-3; MITP-Verlag, Bonn 2003 • Pohlmann, N.; Reimer, H.: "Trusted Computing - Ein Weg zu neuen IT- Sicherheitsarchitekturen”, ISBN 978-3-8348-0309-2, Vieweg-Verlag, Wiesbaden 2008 • H. Blumberg, N. Pohlmann: "Der IT-Sicherheitsleitfaden“, 2. aktualisierte und erweiterte Auflage, ISBN-10: 3-8266-1635-9; 523 Seiten, MITP-Verlag, Bonn 2006 • M. Hertlein, P. Manaras, N. Pohlmann: “Bring Your Own Device For Authentication (BYOD4A) – The Xign–System“. In Proceedings of the ISSE 2015 - Securing Electronic Business Processes - Highlights of the Information Security Solutions Europe 2015 Conference, Eds.: N. Pohlmann, H. Reimer, W. Schneider; Springer Vieweg Verlag, Wiesbaden 2015
Bemerkungen:	keine

Komponentenbasierte Softwareentwicklung

Kürzel:	KBE		
Untertitel:	Komponentenbasierte Softwareentwicklung und Frameworks		
Studiensemester:	(Bachelor)		
Modulverantwortliche(r):	Prof. Jürgen Znotka		
Dozent(in):	Prof. Jürgen Znotka		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	WP	-	WP
Lehrform / SWS:	2 SWS Vorlesung, 2 Praktikum		
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20		
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Wintersemester, nach Bedarf		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	Softwaretechnik, Datenbanken, Internet-Datenbanken, GUI-Programmierung		
Angestrebte Lernergebnisse:	Die Studierenden kennen • Begriffe der komponentenbasierten Softwareentwicklung • Begriffe der speziellen JEE Entwicklung (Session Beans, Singleton, Message-Driven Beans) • Webservices • Begriffe im Kontext von Frameworks (Inversion of Control IoC) • Begriffe der Aspektorientierte Softwareentwicklung • die folgenden Diagramme der UML: Komponentendiagramm, Verteilungsdiagramm • Begriffe der Softwarequalität wie Functionality, Usability, Reliability, Portability und Supportability (FURPS) Die Studierenden verstehen		

	• den Zusammenhang der einzelnen Phasen in verschiedenen Softwareprozessen und die jeweiligen Vor- und Nachteile • den Zusammenhang zwischen Anforderungen und objektorientierten Modellen • Die Studierenden können das Erlernte anwenden, um • aus unstrukturierten Anforderungen an ein System funktionale Anforderungen zu extrahieren • qualitative Anforderungen zu formulieren • objektorientierte Modelle auf Basis der UML zu erstellen für verschiedene Anwendungsdomänen
Inhalt:	• Einführung komponentenbasierte Softwareentwicklung • Java Enterprise Komponentenmodell • Session Beans • Singleton Bean • Message-Driven Beans • Webservices • Aspektorientierte Softwareentwicklung • Einführung in Frameworks • Ein spezielles Framework • UML Diagramme: Komponentendiagramm und Verteilungsdiagramm
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (60 Min.)
Literatur:	• Sommerville, Ian: Software Engineering, Addison-Wesley, 10th Edition, 2015 • George T. Heineman, William T. Councill: Component-Based Software Engineering: Putting the Pieces Together, Addison-Wesley Professional, 2001 • Clemens Szyperski: Component Software: Beyond Object-Oriented Programming. 2nd Edition, Addison-Wesley, 2002 • Eric Jendrock, Ricardo Cervera-Navarro, Ian Evans, Kim Haase, William Markito: The Java EE 7 Tutorial, 2014 • SPRING Framework documentation: https://spring.io/
Bemerkungen:	---

Künstliche Intelligenz

Kürzel:	KI		
Untertitel:	---		
Studiensemester:	4. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr. Wolfram Conen		
Dozent(in):	Prof. Dr. Wolfram Conen		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	WP	-	-
Lehrform / SWS:	3 SWS Vorlesung, 1 SWS Übung		
Gruppengröße:	Standard		
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Sommersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	LDS, ADS, EPR, OPR		
Angestrebte Lernergebnisse:	Die Studierenden kennen die Grundzüge der Entwicklungsgeschichte der Künstlichen Intelligenz (KI). Sie kennen grundlegende Begriffe der Stochastik und des maschinellen Lernens, insbesondere der bayes'schen Modellierung, und können diese anwenden. Sie sind in der Lage, typische Problemsituationen aus den Feldern intelligentes Datamining (Klassifikation, Lernen aus Daten, Bayes'sche Inferenz) und Optimierung rationaler Entscheidungen (insbesondere Planen und Entscheiden bei unsicherem Wissen) zu modellieren und zu lösen. Sie kennen die Grundzüge der Lösung der genannten Probleme unter Verwendung von neuronalen Netzen. Sie können ihre Erkenntnisse auf verwandte Problemstellungen übertragen und sind darauf vorbereitet, sich vertieft mit Spezialgebieten der KI-Anwendung auseinanderzusetzen. Sie wissen um die Problematik der Interpretierbarkeit von Modellen und sind darauf vorbereitet, die		

	inhaltichen und gesellschaftlichen Fragen, die mit dem Einsatz von KI-Modellen und -Systemen verbunden sind, kompetent zu diskutieren.
Inhalt:	Einführendes zur Geschichte der KI und zur Problemlösung mittels intelligenter Agenten. Grundlegendes zur algorithmischen Problemlösung durch exakte und heuristische Suche. Grundlegendes zur Modellierung und Anwendung von Wissen bei Unsicherheit: Bayes'sche Inferenz, Sampling, Filtering, Decision Making und zugehörige Grundlagen. Grundlegendes zu maschinellem Lernen: Kategorisierung (Naive Bayes, kNN, Decision Trees), Clustering, Collaborative Filtering, Time Series Analysis und zugehörige Grundlagen, insbesondere neuronale Netze (NN), Deep-NN, Graph-NN. Grundlegendes zur sequentiellen Optimierung von Entscheidungen: Adversarial Search, MCTS, Dynamic Programming, Reinforcement Learning (RL), Deep-RL Einführendes zur kritischen Diskussion der Interpretierbarkeit von Modellen des maschinellen Lernens und der inhaltlichen und gesellschaftlichen Konsequenzen ihres Einsatzes. Bonusthema unter Mitarbeit der Studierenden mit Gruppenpräsentation, z.B.: KI und Kreativität.
Studien- / Prüfungsleistungen:	Klausur
Literatur:	• Russell, Norvig: Artificial Intelligence, A Modern Approach, Pearson, in der jeweils aktuellen Auflage • Ertel, Grundkurs Künstliche Intelligenz, Springer, in der jeweils aktuellen Auflage • Ergänzende grundlegende und aktuelle Forschungsarbeiten und Vorträge.
Bemerkungen:	---

Knowledge Graphs

Kürzel:	KNG
Untertitel:	-
Studiensemester:	4 (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Katja Zeume
Dozent(in):	Prof. Dr. Katja Zeume
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: nicht begrenzt, Übung: 30, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 56 Zeitstunden Selbststudium: 124 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester (nach Bedarf)
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Über den dazugehörenden Moodle-Kurs
Voraussetzungen nach Prüfungsordnung:	keine
Empfohlene Voraussetzungen (Modulprüfungen):	Datenbanksysteme
Angestrebte Lernergebnisse:	Die Studierenden lernen die praktische Anwendung von „Knowledge Graphs“ in der heutigen IT-Landschaft kennen. Die Studierenden lernen welchen typischen Probleme mit Knowledge Graphs gelöst werden und welche Probleme dabei auftreten können. Die Studierenden lernen den Umgang an einer praktischen Graph-Datenbank Implementierung (z.Bsp. RDF, SPARQL) kennen. Dabei lernen die Studierenden ihre Kenntnisse über relationale Datenbanksysteme auf eine erste nicht-relationale Technologie zu erweitern.
Inhalt:	Die Veranstaltung bietet eine Einführung in das Thema „Knowledge Graphs“ im Kontext der

	Vertiefung von nicht-relationalen Datenbankformaten. - Einführung in das Thema „Knowledge Graphs“ anhand von aktuellen Beispielanwendungen bzw. Problemfeldern. - Praktische Einführung einer Graph-Datenbank (z. Bsp. RDF und SPARQL). - Überblick Schemasprachen für Graphen. - Überblick Anfragesprachen für Graph-Datenbanken und deren spezielle Problemstellungen. - (Wahlweise) Weitere Technologien zum Thema und der Vergleich von Vor- und Nachteilen.
Studien- / Prüfungsleistungen:	Klausur oder mündliche Prüfung
Literatur:	P.Hitzler, M. Krötzsch, S. Rudolph: Foundations of Semantic Web Technologies, CRC Press, 2009. T. Heath, Ch. Bitzer: Linked Data – Evolving the Web into a Global Data Space, Morgan & Claypool, 2011.
Bemerkungen:	-

Mobile und Cloud Computing

Kürzel:	MCC
Untertitel:	
Studiensemester:	4. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Detlef Mansel
Dozent(in):	Prof. Dr. Detlef Mansel
Sprache:	deutsch
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum
Gruppengröße:	Vorlesung: nicht begrenzt, Übung 30, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 56 Zeitstunden Selbststudium: 124 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung für Übung und Praktikum via Moodle
Voraussetzungen nach Prüfungsordnung:	keine
Empfohlene Voraussetzungen (Modulprüfungen):	Modul Rechnernetze
Angestrebte Lernergebnisse:	• Die Studierenden kennen grundlegende Cloud Technologien und deren Eigenschaften • Die Studierenden verstehen die enorme Bedeutung einer performanten Netzwerkanbindung. • Die Studierenden erwerben in Ergänzung zu den im Modul Rechnernetzen erworbenen Kompetenzen zum Umgang mit Festnetzen Fähigkeiten zum Umgang mit den für mobile Anwendungen verwendeten relevanten Mobilfunksystemen. • Sie können grundlegend mit den Einschränkungen der Funkanbindung mobiler Endgeräte umgehen und darauf aufbauend beurteilen, welchen Einfluss diese Einschränkungen auf die Effizienz der von Ihnen zu verantwortenden Software haben.
Inhalt:	Grundlagen zu Cloud Computing und XaaS Typen mobiler Netze• Bluetooth als Beispiel für ein Ad hoc Netz• GSM/UMTSLTE als zellulares Infrastruktur-Netz• Wireless LAN (WLAN) •LoRaWAN als IoT Netz.

	Praktikum mit Versuchen zu ausgewählten Funksystemen
Studien- / Prüfungsleistungen:	Studienleistungen: Die Studierenden können während des Praktikums Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der Vorlesung bekannt gegeben. Prüfungsleistungen: Klausur oder mündliche Prüfung
Literatur:	• Jens Riwotzki, Cloud-Computing Theorie und Praxis, HERDT-Verlag • Ulrich Trick, Einführung in die Mobilfunknetze der 5. Generation, Walter de Gruyter GmbH • Michael Sauter, Grundkurs Mobile Kommunikationssysteme, Springer Vieweg, aktuellste Auflage • Aktuelle Ergänzungen im Moodle-Kurs zu diesem Modul
Bemerkungen:	-

Mobile Robotik

Kürzel:	MRO		
Untertitel:	Grundlegende Konzepte und Verfahren zur Entwicklung von Applikationen für mobile Roboter		
Studiensemester:	5. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr.-Ing. Dipl. Inform. Hartmut Surmann		
Dozent(in):	Prof. Dr.-Ing. Dipl. Inform. Hartmut Surmann		
Sprache:	Deutsch, englisch bei Bedarf		
Zuordnung zum Curriculum:	IN	ID	WI
	WP	-	WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum		
Gruppengröße:	Standard		
Arbeitsaufwand:	Kontaktzeit: 75 Zeitstunden Selbststudium: 105 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Wintersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Keine		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Objektorientierte Programmierung, Algorithmen und Datenstrukturen, Mathematische Grundlagen, Statistik und Lineare Algebra		
Angestrebte Lernergebnisse:	Die Studierenden lernen die Begriffe und Komponenten von mobilen Robotern sowie die Konzepte und Methoden der Programmierung kennen und können diese effektiv und strukturiert bei der Entwicklung eigener Steuerungsprogramme einsetzen. Sie lernen wie unterschiedliche Sensordaten fusioniert werden und mobile Systeme navigieren sowie sich selbst lokalisieren. Sie kennen die Gefahren beim Umgang mit mobilen Systemen und die Wichtigkeit der Einhaltung von Vorschriften sowohl auf technischer als auch sozialer Ebene. Neben der Programmiermethodik lernen die Studierenden die Verwendung von weiteren Bibliotheken des Roboter Frameworks ROS (Robot Operation System) kennen.		

Inhalt:	• Roboterprogrammierung, Roboterframework ROS, • Sensorik • Aktuatorik • Lokalisierung • Kartenbau • Navigation • Planung • Betriebssystem: Linux + ROS; Lehrsprache ist C / C++, Python, ipython notebooks.
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• J. Hertzberg, K. Lingemann, A. Nüchter: „Mobile Roboter: Eine Einführung aus Sicht der Informatik“, ISBN 978-3642017254 • B. Siciliano, O. Khatib (Eds.): „Handbook of Robotic“, ISBN 978-3-540-23957-4 • Craig, J.J. (2004), „Introduction to Robotics: Mechanics and Control (3rd Edition)“, 8, 2004. Prentice Hall • R. Siegwart „Introduction to Autonomous Mobile Robots“, MIT Press, ISBN: 978-0-262-19502 -7 • S. Thrun, W. Burgard, D. Fox: „Probabilistic Robotics“, ISBN 978-0262201629
Bemerkungen:	https://moodle.w-hs.de/course/index.php?categoryid=232

Parallele Programmierung

Kürzel:	PAP
Untertitel:	Grundlegende Methoden und Algorithmen paralleler Programmierung
Studiensemester:	4 (Bachelor)
Modulverantwortliche(r):	Prof. Dr.-Ing. Dipl. Inform. Hartmut Surmann
Dozent(in):	Prof. Dr.-Ing. Dipl. Inform. Hartmut Surmann
Sprache:	deutsch, englisch bei Bedarf
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: -
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum
Gruppengröße:	Standard
Arbeitsaufwand:	Kontaktzeit: 70 Zeitstunden Selbststudium: 110 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Keine
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Objektorientierte Programmierung, Algorithmen und Datenstrukturen, Theoretische Informatik
Angestrebte Lernergebnisse:	Die Studierenden lernen die Grundlagen und Begriffe der parallelen Programmierung und des parallelen Programmierparadigma kennen und können parallele Programme entwickeln und testen. Sie lernen sequentielle Algorithmen zu parallelisieren und innerhalb der Grafikkarte oder MultiCore Architekturen oder über mehrere Rechner hinweg parallel zu verteilen. Neben der Programmiermethodik, parallelen Pattern und dem Design lernen die Studierenden die speziellen Probleme und Fragestellungen bei der parallelen Programmierung kennen, insbesondere das Erkennen von Nebenläufigkeiten und die schwierigere Fehleranalyse.
Inhalt:	• Grundlagen paralleler Programmierung • Parallele Architekturen • Design und Analyse von parallelen Algorithmen

	• Threads- OpenMP • MPI - OpenCL • CUDA- Parallele Pattern (Map, Reduce, Scan, Sort, ...)
Studien- / Prüfungsleistungen:	Prüfungsleistungen: Mündliche Prüfung (30 Min.) oder Klausur (90 Min.) je nach Teilnehmerzahl (>12 Klausur)
Literatur:	• Thomas Rauber: “Parallele Programmierung”, Springer Verlag, ISBN 978-3-540-46549-2.- S. Hoffmann, R. Lienhart: "OpenMP" • T. Rauber, G. Rünger: "Multicore: Parallele Programmierung" • Norm Matloff: "Programming on Parallel Machines; GPU, Multicore, Clusters and More"
Bemerkungen:	https://moodle.w-hs.de/course/index.php?categoryid=232

Prozedurale Programmierung

Kürzel:	PPR		
Untertitel:	---		
Studiensemester:	5. (Bachelor)		
Modulverantwortliche(r):	Prof. Dr. Ulrike Griefahn		
Dozent(in):	Lehrbeauftragte/r		
Sprache:	Deutsch		
Zuordnung zum Curriculum:	IN	ID	WI
	WP	-	-
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 1 SWS Praktikum		
Gruppengröße:	Vorlesung: nicht begrenzt, Übung: 40, Praktikum: 20		
Arbeitsaufwand:	Kontaktzeit: 60 Zeitstunden Selbststudium: 120 Zeitstunden		
Leistungspunkte:	6		
Turnus:	Wintersemester, jährlich		
Teilnehmerzahl:	Nicht begrenzt		
Anmeldungsmodalitäten:	Vorlesung: keine, Übungen und Praktikum: über Moodle-Kurs		
Voraussetzungen nach Prüfungsordnung:	Keine modulspezifischen Voraussetzungen		
Empfohlene Voraussetzungen (Modulprüfungen):	Einführung in die Programmierung, Objektorientierte Programmierung, Algorithmen und Datenstrukturen		
Angestrebte Lernergebnisse:	Die Studierenden kennen die Konzepte und Methoden der prozeduralen Programmierung und können diese effektiv und strukturiert bei der Entwicklung eigener prozeduraler Programme mit der Programmiersprache C einsetzen. Sie gehen sicher mit maschinennahen Konzepten wie Zeigern und Speicherverwaltung sowie mit Strukturen um. Die Studierenden sind damit in der Lage, sich zukünftig selbstständig und zügig in weitere prozedurale Sprachen einzuarbeiten.		
Inhalt:	• Grundelemente von C • Funktionen und Speicherklassen • Präprozessor • Adressen und Zeiger • Dynamische Speicherverwaltung • Strukturen • Weitere ausgewählte Sprachelemente		

	• Make • Überblick über die Erweiterungen von C zu C++
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (90 Min.) Die Studierenden können während des Praktikums Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der 1. Vorlesungsstunde bekannt gegeben.
Literatur:	• Manfred Dausmann, Ulrich Bröckl und Joachim Goll, C als erste Programmiersprache. Vom Einsteiger zum Profi. 8. überarb. und erw. Auflage, Springer Vieweg, 2014, 727 Seiten, ISBN-13: 978-3-834-81858-4 • Jürgen Wolf, C von A bis Z. Rheinwerk Computing, 3., aktualisierte und erweiterte Auflage, 2009, 1190 Seiten, ISBN-13: 978-3-8362-1411-7 • Vogt: C für Java-Programmierer, Carl Hanser Verlag 2007, 256 Seiten, ISBN-13: 978-3-446-4079-78
Bemerkungen:	---

Practical Security Attacks and Exploitation

Kürzel:	PRAX
Untertitel:	
Studiensemester:	4. (Bachelor)
Modulverantwortliche(r):	Prof. Dr. Christian Dietrich
Dozent(in):	Prof. Dr. Christian Dietrich
Sprache:	deutsch oder englisch
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: WP
Lehrform / SWS:	1 SWS Vorlesung, 3 SWS Praktikum
Gruppengröße:	Vorlesung: nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 56 Zeitstunden Selbststudium: 124 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung via Moodle
Voraussetzungen nach Prüfungsordnung:	keine
Empfohlene Voraussetzungen (Modulprüfungen):	EPR, OPR
Angestrebte Lernergebnisse:	• Verständnis gängiger Verfahren zur Systemsicherheit, Systemintegrität und zum Softwareschutz • Anwenden von Mechanismen zur Identifikation und Ausnutzung von Software-Schwachstellen • Anwenden von Angriffstechniken in Computernetzwerken • Erlangen von Kenntnissen im Bereich der Schadsoftware-Erkennung und -Abwehr • Teilnahme an einem Capture-the-Flag-Wettbewerb
Inhalt:	Die Studierenden lernen die Anwendbarkeit und Grenzen von sicherheitsrelevanten Angriffen gegen Systeme, Netzwerkprotokolle und Software. Dabei werden die folgenden Themen behandelt: • Linux and Unix-like operating system basics • Vulnerability research

	• Reconnaissance and scanning • System security and operational security • Software security • Bytecode and binary code analysis • Denial-of-Service attacks • Web security • Incident response Lerneinheiten bestehen jeweils aus einer Einführung in Form mindestens einer Vorlesungseinheit sowie Aufgaben, die im Praktikum gelöst werden müssen. Darüber hinaus müssen die Studierenden selbst verwundbare Beispiele als Aufgaben entwerfen, die beispielsweise im Rahmen eines eigenen CTF-Wettbewerbs eingesetzt werden könnten.
Studien- / Prüfungsleistungen:	Studienleistungen: Die Studierenden können während des Praktikums Bonuspunkte für die Klausur erwerben. Einzelheiten zum Erwerb der Bonuspunkte werden in der Vorlesung bekannt gegeben. Prüfungsleistungen: Ausarbeitung und/oder Klausur oder mündliche Prüfung
Literatur:	• Eckert, C.: *IT-Sicherheit. Konzepte, Verfahren, Protokolle.* Oldenbourg, München, aktuellste Auflage • Erickson, J.: *Hacking - The Art of Exploitation*. No Starch Press; aktuellste Auflage • Aktuelle wissenschaftliche Publikationen
Bemerkungen:	-

Software Design

Kürzel:	SWD
Untertitel:	Architektur und Design komplexer Softwaresysteme
Studiensemester:	(Bachelor)
Modulverantwortliche(r):	Prof. Jürgen Znotka
Dozent(in):	Prof. Jürgen Znotka
Sprache:	Deutsch
Zuordnung zum Curriculum:	IN: WP; ID: -; WI: WP
Lehrform / SWS:	2 SWS Vorlesung, 1 SWS Übung, 2 SWS Praktikum
Gruppengröße:	Vorlesung: Nicht begrenzt, Praktikum: 20
Arbeitsaufwand:	Kontaktzeit: 70 Zeitstunden Selbststudium: 110 Zeitstunden
Leistungspunkte:	6
Turnus:	Sommersemester, jährlich
Teilnehmerzahl:	Nicht begrenzt
Anmeldungsmodalitäten:	Anmeldung über den Moodle-Kurs zu diesem Modul
Voraussetzungen nach Prüfungsordnung:	Keine
Empfohlene Voraussetzungen (Modulprüfungen):	Softwaretechnik, Datenbanksysteme, Data on the Web
Angestrebte Lernergebnisse:	Die Studierenden kennen • Architekturmuster • Designmuster • OSGi Komponentenmodell Die Studierenden verstehen • den Zusammenhang der einzelnen Phasen in verschiedenen Softwareprozessen und die jeweiligen Vor- und Nachteile, insbesondere den Übergang von Analyse zu Design Die Studierenden können das Erlernte anwenden, um

	• aus einem Pflichtenheft ein Design zu entwickeln • qualitative Anforderungen an das Design zu formulieren • objektorientierte Designmodelle auf Basis der UML zu erstellen
Inhalt:	• Einführung Software Design • Design Patterns (Observer, Adapter, Fassade, Strategie, Dekorierer, Simple Fabrik, Fabrikmethode, abstrakte Fabrik, Watchdog) • Einführung in Architekturmuster • MVC (Model-View-Controller) und dessen Derivate Passive View und Supervising Controller • Mehrschichtarchitektur • UML Diagramme: Interaktionsübersicht, Kommunikationsdiagramm, Paketdiagramm, Kompositionsstrukturdiagramm, Komponentendiagramm, Verteilungsdiagramm) • Komponentenmodell OSGi
Studien- / Prüfungsleistungen:	Studienleistungen laut Prüfungsordnung als Voraussetzung zur Prüfungsteilnahme: Keine Prüfungsleistungen: Klausur (60 Min.)
Literatur:	• Sommerville, Ian: Software Engineering, Addison-Wesley, 10th Edition, 2015 • Fowler, Martin: Patterns of Enterprise Application Architecture, Addison-Wesley, 2002 • Rup, Chris u.a. UML 2 glasklar: Praxiswissen für die UML-Modellierung, Hanser, 4. Auflage, 2012 • Kirk Knoernschild: Java Application Architecture: Modularity Patterns with Examples Using OSGi, Prentice Hall, 2012
Bemerkungen:	---