
Modulhandbuch

M.Sc. Informatik, PO 2018

Fakultät für Angewandte Informatik

Wintersemester 2020/2021

Studienbeginn ab Wintersemester 2018/2019

Wichtige Zusatzinformation für das WS 2020/21 aufgrund der Corona-Pandemie:

Bitte berücksichtigen Sie, dass aufgrund der Entwicklungen der Corona-Pandemie die Angaben zu den jeweiligen Prüfungsformaten in den Modulhandbüchern ggf. noch nicht aktuell sind. Welche Prüfungsformate schließlich bei welchen Modulen möglich sein werden, wird im weiteren Verlauf des Semesters geklärt und festgelegt werden. Entsprechende Informationen werden spätestens am 01.12.2020 bekannt gegeben.

Liebe Studierenden,

auch wenn COVID-19 uns vor besondere Herausforderungen stellt, tun wir unser bestes, um euch weiterhin ein breit aufgestelltes Lehrangebot zu bieten. Ihr findet also ähnlich viele Veranstaltungen im Modulhandbuch wie in einem "normalen" Semester auch.

Im Lehrangebot haben sich jedoch einige Änderungen ergeben:

1. Durch die Pensionierung von Prof. Knorr zum Ende des Sommersemesters 2020 fallen die Module *INF-0084: Seminar Next Generation Networks* und *INF-0085: Projektmodul Kommunikationssysteme* ersatzlos weg.
2. Als Nachfolger von Prof. Möller wurde Prof. Mömke berufen. Er bietet die Module *INF-0371: Approximation Algorithms* und *INF-0374: Projektmodul Resource Aware Algorithmics* neu an, weitere werden im SoSe 2021 folgen.
3. Außerdem sind folgende Module neu hinzu gekommen:
 - INF-0364: Seminar Software Engineering in sicherheitskritischen Systemen (MA)
 - INF-0367: Advanced Machine Learning and Computer Vision
 - INF-0368: Embedded Hardware

Da das Modulhandbuch ein Service für euch als Studierende ist, arbeiten wir eng mit der Studierendenvertretung Informatik zusammen. Solltet Ihr Anregungen, Fragen, Kritik oder Verbesserungsvorschläge zum neuen Modulhandbuch haben, so teilt diese einfach der Studierendenvertretung Informatik mit. Ihr erreicht sie unter hallo@fachschaft-info.de und persönlich im Raum 1007N.

Viele Grüße,

Euer Modulhandbuch-Beauftragter

Martin Frieb

Übersicht nach Modulgruppen

1) Theoretische Informatik (ECTS: 5)

Module im Umfang von 5 LP sind zu erbringen.

INF-0051: Algorithmen für NP-harte Probleme (8 ECTS/LP, Wahlpflicht).....	9
INF-0052: Einführung in die Komplexitätstheorie (5 ECTS/LP, Wahlpflicht).....	11
INF-0053: I/O-effiziente Algorithmen (5 ECTS/LP, Wahlpflicht).....	13
INF-0054: Datenstrukturen (8 ECTS/LP, Wahlpflicht) *	15
INF-0056: Online-Algorithmen (5 ECTS/LP, Wahlpflicht).....	17
INF-0130: Formale Methoden im Software Engineering (8 ECTS/LP, Wahlpflicht).....	19
INF-0156: Algebraische Beschreibung paralleler Prozesse (6 ECTS/LP, Wahlpflicht).....	21
INF-0157: Endliche Automaten (5 ECTS/LP, Wahlpflicht) *	23
INF-0161: Petrinetze - eine Theorie paralleler Systeme (5 ECTS/LP, Wahlpflicht).....	25
INF-0163: Verteilte Algorithmen (8 ECTS/LP, Wahlpflicht) *	27
INF-0201: Platzeffiziente Algorithmen (5 ECTS/LP, Wahlpflicht).....	29
INF-0243: Process Mining (6 ECTS/LP, Wahlpflicht).....	31
INF-0310: Perlen der Algorithmik (5 ECTS/LP, Wahlpflicht).....	33
INF-0371: Approximation Algorithms (5 ECTS/LP, Wahlpflicht) *	35

2) Praktische Informatik (ECTS: 5)

Module im Umfang von 5 LP sind zu erbringen.

INF-0037: Praktikum Automotive Software Engineering (6 ECTS/LP, Wahlpflicht).....	37
INF-0066: Organic Computing II (5 ECTS/LP, Wahlpflicht).....	39
INF-0077: Suchmaschinen (8 ECTS/LP, Wahlpflicht).....	41
INF-0078: Datenbankprogrammierung (Oracle) (5 ECTS/LP, Wahlpflicht) *	43
INF-0088: Bayesian Networks (5 ECTS/LP, Wahlpflicht).....	45
INF-0093: Probabilistic Robotics (5 ECTS/LP, Wahlpflicht).....	47
INF-0129: Softwaretechnik II (8 ECTS/LP, Wahlpflicht) *	49
INF-0130: Formale Methoden im Software Engineering (8 ECTS/LP, Wahlpflicht).....	52
INF-0131: Software- und Systemsicherheit (8 ECTS/LP, Wahlpflicht).....	54
INF-0133: Selbstorganisierende, adaptive Systeme (8 ECTS/LP, Wahlpflicht) *	56

* = Im aktuellen Semester wird mindestens eine Lehrveranstaltung für dieses Modul angeboten

INF-0179: Einführung in die Spieleprogrammierung (8 ECTS/LP, Wahlpflicht).....	58
INF-0182: Praktikum Multimodale Echtzeitsignalverarbeitung (8 ECTS/LP, Wahlpflicht).....	60
INF-0183: Praktikum Spieleprogrammierung (8 ECTS/LP, Wahlpflicht) *.....	61
INF-0189: Qualitätssicherung im Software Engineering (5 ECTS/LP, Wahlpflicht).....	63
INF-0207: Reinforcement Learning (8 ECTS/LP, Wahlpflicht) *.....	65
INF-0213: Parallele und Verteilte Datenbanksysteme (5 ECTS/LP, Wahlpflicht).....	67
INF-0217: Praktikum Autonomes Fahren (10 ECTS/LP, Wahlpflicht).....	69
INF-0233: Industrierobotik (8 ECTS/LP, Wahlpflicht) *.....	71
INF-0235: Software für Industrie 4.0 (5 ECTS/LP, Wahlpflicht).....	73
INF-0248: Kollaborative Robotik (8 ECTS/LP, Wahlpflicht) *.....	75
INF-0250: Praktikum Reinforcement Learning (8 ECTS/LP, Wahlpflicht).....	78
INF-0272: Intelligente Signalanalyse in der Medizin (5 ECTS/LP, Wahlpflicht).....	79
INF-0273: Praktikum Mobile Sensing for Fitness and Wellbeing (5 ECTS/LP, Wahlpflicht).....	81
INF-0277: Analyzing Massive Data Sets (8 ECTS/LP, Wahlpflicht).....	83
INF-0279: Music Informatics (5 ECTS/LP, Wahlpflicht).....	85
INF-0284: Praktikum Mobile Application Development (5 ECTS/LP, Wahlpflicht).....	87
INF-0291: Praktikum Selbstlernende Systeme (8 ECTS/LP, Wahlpflicht) *.....	89
INF-0293: Advanced Deep Learning (8 ECTS/LP, Wahlpflicht) *.....	91
INF-0294: Speech Pathology (5 ECTS/LP, Wahlpflicht) *.....	93
INF-0296: Praktikum Interactive Machine Learning (8 ECTS/LP, Wahlpflicht) *.....	95
INF-0307: Modellbasierte Entwicklung und Analyse von Software Systemen (6 ECTS/LP, Wahlpflicht).....	97
INF-0308: Software-intensive Systeme (6 ECTS/LP, Wahlpflicht) *.....	99
INF-0315: Deep Learning (5 ECTS/LP, Wahlpflicht).....	101
INF-0316: Machine Learning and Computer Vision (8 ECTS/LP, Wahlpflicht).....	103
INF-0318: Praktikum Simulation von autonomen Fahrzeugen (Vertiefung) (8 ECTS/LP, Wahlpflicht) *.....	105
INF-0353: Praktikum Programmieren in der biomedizinischen Informatik (Master) (6 ECTS/LP, Wahlpflicht).....	107
INF-0367: Advanced Machine Learning and Computer Vision (5 ECTS/LP, Wahlpflicht) *.....	109
ZCS-2000: Softskills (2 ECTS/LP, Wahlpflicht) *.....	111

3) Technische Informatik (ECTS: 5)

* = Im aktuellen Semester wird mindestens eine Lehrveranstaltung für dieses Modul angeboten

Module im Umfang von 5 LP sind zu erbringen.

INF-0067: Peer-to-Peer und Cloud Computing (5 ECTS/LP, Wahlpflicht).....	121
INF-0069: Weiterführende Betriebssystemkonzepte (8 ECTS/LP, Wahlpflicht).....	123
INF-0147: Prozessorarchitektur (5 ECTS/LP, Wahlpflicht).....	125
INF-0149: Praktikum Eingebettete Systeme (5 ECTS/LP, Wahlpflicht).....	127
INF-0150: Hardware-Entwurf (8 ECTS/LP, Wahlpflicht).....	129
INF-0216: Vertiefte Multicore-Programmierung (8 ECTS/LP, Wahlpflicht).....	131
INF-0297: Praktikum Prozessorbau (5 ECTS/LP, Wahlpflicht).....	133
INF-0309: Echtzeitsysteme (8 ECTS/LP, Wahlpflicht) *	135
INF-0319: Praktikum Interdisziplinäres Projekt Ingenieurinformatik (6 ECTS/LP, Wahlpflicht).....	137
INF-0335: Safety-Critical Systems (5 ECTS/LP, Wahlpflicht).....	138
INF-0368: Embedded Hardware (5 ECTS/LP, Wahlpflicht) *	140

4) Wahlbereich Informatik (ECTS: 61)

Module im Umfang von 61 LP sind zu erbringen.

INF-0037: Praktikum Automotive Software Engineering (6 ECTS/LP, Wahlpflicht).....	142
INF-0051: Algorithmen für NP-harte Probleme (8 ECTS/LP, Wahlpflicht).....	144
INF-0052: Einführung in die Komplexitätstheorie (5 ECTS/LP, Wahlpflicht).....	146
INF-0053: I/O-effiziente Algorithmen (5 ECTS/LP, Wahlpflicht).....	148
INF-0054: Datenstrukturen (8 ECTS/LP, Wahlpflicht) *	150
INF-0056: Online-Algorithmen (5 ECTS/LP, Wahlpflicht).....	152
INF-0066: Organic Computing II (5 ECTS/LP, Wahlpflicht).....	154
INF-0067: Peer-to-Peer und Cloud Computing (5 ECTS/LP, Wahlpflicht).....	156
INF-0069: Weiterführende Betriebssystemkonzepte (8 ECTS/LP, Wahlpflicht).....	158
INF-0077: Suchmaschinen (8 ECTS/LP, Wahlpflicht).....	160
INF-0078: Datenbankprogrammierung (Oracle) (5 ECTS/LP, Wahlpflicht) *	162
INF-0088: Bayesian Networks (5 ECTS/LP, Wahlpflicht).....	164
INF-0093: Probabilistic Robotics (5 ECTS/LP, Wahlpflicht).....	166
INF-0129: Softwaretechnik II (8 ECTS/LP, Wahlpflicht) *	168
INF-0130: Formale Methoden im Software Engineering (8 ECTS/LP, Wahlpflicht).....	171
INF-0131: Software- und Systemsicherheit (8 ECTS/LP, Wahlpflicht).....	173

INF-0133: Selbstorganisierende, adaptive Systeme (8 ECTS/LP, Wahlpflicht) *	175
INF-0147: Prozessorarchitektur (5 ECTS/LP, Wahlpflicht)	177
INF-0149: Praktikum Eingebettete Systeme (5 ECTS/LP, Wahlpflicht)	179
INF-0150: Hardware-Entwurf (8 ECTS/LP, Wahlpflicht)	181
INF-0156: Algebraische Beschreibung paralleler Prozesse (6 ECTS/LP, Wahlpflicht)	183
INF-0157: Endliche Automaten (5 ECTS/LP, Wahlpflicht) *	185
INF-0161: Petrinetze - eine Theorie paralleler Systeme (5 ECTS/LP, Wahlpflicht)	187
INF-0163: Verteilte Algorithmen (8 ECTS/LP, Wahlpflicht) *	189
INF-0179: Einführung in die Spieleprogrammierung (8 ECTS/LP, Wahlpflicht)	191
INF-0182: Praktikum Multimodale Echtzeitsignalverarbeitung (8 ECTS/LP, Wahlpflicht)	193
INF-0183: Praktikum Spieleprogrammierung (8 ECTS/LP, Wahlpflicht) *	194
INF-0189: Qualitätssicherung im Software Engineering (5 ECTS/LP, Wahlpflicht)	196
INF-0201: Platzeffiziente Algorithmen (5 ECTS/LP, Wahlpflicht)	198
INF-0207: Reinforcement Learning (8 ECTS/LP, Wahlpflicht) *	200
INF-0213: Parallele und Verteilte Datenbanksysteme (5 ECTS/LP, Wahlpflicht)	202
INF-0216: Vertiefte Multicore-Programmierung (8 ECTS/LP, Wahlpflicht)	204
INF-0217: Praktikum Autonomes Fahren (10 ECTS/LP, Wahlpflicht)	206
INF-0233: Industrierobotik (8 ECTS/LP, Wahlpflicht) *	208
INF-0235: Software für Industrie 4.0 (5 ECTS/LP, Wahlpflicht)	210
INF-0243: Process Mining (6 ECTS/LP, Wahlpflicht)	212
INF-0248: Kollaborative Robotik (8 ECTS/LP, Wahlpflicht) *	214
INF-0250: Praktikum Reinforcement Learning (8 ECTS/LP, Wahlpflicht)	217
INF-0272: Intelligente Signalanalyse in der Medizin (5 ECTS/LP, Wahlpflicht)	218
INF-0273: Praktikum Mobile Sensing for Fitness and Wellbeing (5 ECTS/LP, Wahlpflicht)	220
INF-0277: Analyzing Massive Data Sets (8 ECTS/LP, Wahlpflicht)	222
INF-0279: Music Informatics (5 ECTS/LP, Wahlpflicht)	224
INF-0284: Praktikum Mobile Application Development (5 ECTS/LP, Wahlpflicht)	226
INF-0291: Praktikum Selbstlernende Systeme (8 ECTS/LP, Wahlpflicht) *	228
INF-0293: Advanced Deep Learning (8 ECTS/LP, Wahlpflicht) *	230
INF-0294: Speech Pathology (5 ECTS/LP, Wahlpflicht) *	232
INF-0296: Praktikum Interactive Machine Learning (8 ECTS/LP, Wahlpflicht) *	234

INF-0297: Praktikum Prozessorbau (5 ECTS/LP, Wahlpflicht).....	236
INF-0307: Modellbasierte Entwicklung und Analyse von Software Systemen (6 ECTS/LP, Wahlpflicht).....	238
INF-0308: Software-intensive Systeme (6 ECTS/LP, Wahlpflicht) *.....	240
INF-0309: Echtzeitsysteme (8 ECTS/LP, Wahlpflicht) *.....	242
INF-0310: Perlen der Algorithmik (5 ECTS/LP, Wahlpflicht).....	244
INF-0315: Deep Learning (5 ECTS/LP, Wahlpflicht).....	246
INF-0316: Machine Learning and Computer Vision (8 ECTS/LP, Wahlpflicht).....	248
INF-0319: Praktikum Interdisziplinäres Projekt Ingenieurinformatik (6 ECTS/LP, Wahlpflicht).....	250
INF-0335: Safety-Critical Systems (5 ECTS/LP, Wahlpflicht).....	251
INF-0353: Praktikum Programmieren in der biomedizinischen Informatik (Master) (6 ECTS/LP, Wahlpflicht).....	253
INF-0367: Advanced Machine Learning and Computer Vision (5 ECTS/LP, Wahlpflicht) *.....	255
INF-0368: Embedded Hardware (5 ECTS/LP, Wahlpflicht) *.....	257
INF-0371: Approximation Algorithms (5 ECTS/LP, Wahlpflicht) *.....	259
ZCS-2000: Softskills (2 ECTS/LP, Wahlpflicht) *.....	261

5) Seminar (ECTS: 4)

Module im Umfang von 4 LP sind zu erbringen.

INF-0058: Seminar Algorithmen und Datenstrukturen für Master (4 ECTS/LP, Wahlpflicht).....	271
INF-0070: Seminar Organic Computing (4 ECTS/LP, Wahlpflicht) *.....	272
INF-0071: Seminar Naturalanaloge Algorithmen und Multiagentensysteme (4 ECTS/LP, Wahlpflicht).....	274
INF-0095: Seminar Multimedia Computing (MA) (4 ECTS/LP, Wahlpflicht) *.....	276
INF-0136: Seminar Software- und Systems Engineering (Master) (4 ECTS/LP, Wahlpflicht) *.....	278
INF-0164: Seminar Theorie verteilter Systeme A (4 ECTS/LP, Wahlpflicht).....	280
INF-0227: Seminar Datenbanksysteme für Master (4 ECTS/LP, Wahlpflicht).....	282
INF-0232: Seminar Medical Information Sciences (MA) (4 ECTS/LP, Wahlpflicht) *.....	284
INF-0240: Seminar Informationssysteme für Master (4 ECTS/LP, Wahlpflicht) *.....	286
INF-0251: Seminar Artificial Intelligence (4 ECTS/LP, Wahlpflicht).....	288
INF-0274: Seminar Embedded Intelligence for Health Care and Wellbeing (Master) (4 ECTS/LP, Wahlpflicht).....	290
INF-0314: Seminar IT-Infrastrukturen in der Medizin für Master (4 ECTS/LP, Wahlpflicht) *.....	292

INF-0320: Seminar Process Mining (4 ECTS/LP, Wahlpflicht).....	294
INF-0331: Seminar Computational Intelligence (Master) (4 ECTS/LP, Wahlpflicht) *	295
INF-0337: Seminar Embedded Systems (Master) (4 ECTS/LP, Wahlpflicht) *	297
INF-0342: Seminar Digital Health (Master) (4 ECTS/LP, Wahlpflicht) *	299
INF-0344: Seminar Software Engineering verteilter Systeme (MA) (4 ECTS/LP, Wahlpflicht) *	301
INF-0346: Seminar Automotive Software and Systems Engineering (MA) (4 ECTS/LP, Wahlpflicht) *	303
INF-0348: Seminar Avionic Software and Systems Engineering (MA) (4 ECTS/LP, Wahlpflicht).....	305
INF-0349: Seminar Menschzentrierte Künstliche Intelligenz (4 ECTS/LP, Wahlpflicht) *	307
INF-0364: Seminar Software Engineering in sicherheitskritischen Systemen (MA) (4 ECTS/LP, Wahlpflicht) *	308

6) Projekt (ECTS: 10)

Module im Umfang von 10 LP sind zu erbringen.

INF-0042: Projektmodul Softwaremethodiken für verteilte Systeme (10 ECTS/LP, Wahlpflicht) *	310
INF-0059: Projektmodul Theoretische Informatik (10 ECTS/LP, Wahlpflicht) *	311
INF-0072: Projektmodul Organic Computing (10 ECTS/LP, Wahlpflicht) *	312
INF-0080: Projektmodul Datenbanken und Informationssysteme (10 ECTS/LP, Wahlpflicht) *	314
INF-0096: Projektmodul Multimedia Computing (10 ECTS/LP, Wahlpflicht) *	316
INF-0108: Projektmodul Lehrprofessur für Informatik (10 ECTS/LP, Wahlpflicht) *	318
INF-0137: Projektmodul Software- und Systems Engineering (10 ECTS/LP, Wahlpflicht) *	320
INF-0165: Projektmodul Theorie verteilter Systeme (10 ECTS/LP, Wahlpflicht) *	322
INF-0170: Projektmodul Human-Centered Multimedia (10 ECTS/LP, Wahlpflicht) *	323
INF-0275: Projektmodul Embedded Intelligence for Health Care and Wellbeing (10 ECTS/LP, Wahlpflicht) *	324
INF-0328: Projektmodul IT-Infrastrukturen in der Medizin (10 ECTS/LP, Wahlpflicht) *	326
INF-0340: Projektmodul Embedded Systems (10 ECTS/LP, Wahlpflicht) *	328
INF-0374: Projektmodul Resource Aware Algorithmics (10 ECTS/LP, Wahlpflicht) *	330

7) Abschlussleistung (ECTS: 30)

Module im Umfang von 30 LP sind zu erbringen.

INF-0003: Masterarbeit (30 ECTS/LP, Wahlpflicht).....	332
---	-----

8) Freiwillige Veranstaltungen

Inhaltsverzeichnis

Die hier aufgeführten Veranstaltungen sind freiwillig und geben keine Leistungspunkte. Ihre Inhalte sind jedoch eine sinnvolle Ergänzung zum bestehenden Lehrangebot.

INF-0221: Anleitung zu wissenschaftlichen Arbeiten (0 ECTS/LP, Wahlfach) *	334
INF-0222: Oberseminar Informatik (0 ECTS/LP, Wahlfach) *	335

Modul INF-0051: Algorithmen für NP-harte Probleme <i>Algorithms for NP-Hard Problems</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
Lernziele/Kompetenzen: Kenntnis verschiedener algorithmischer Lösungsansätze für NP-harte Probleme und die Fähigkeit, diese sinnvoll im Kontext neuer Probleme einzusetzen, zu analysieren und zu bewerten.		
Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.		
Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)		
Voraussetzungen: Empfehlenswert: Gutes Verständnis des Informatik III-Stoffes, insbesondere im Bereich der Graphenalgorithmen. Modul Informatik 3 (INF-0111) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Algorithmen für NP-harte Probleme (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 4		
Inhalte: NP-harte Probleme können nach heutigem Wissen nicht in polynomieller Zeit auf einem üblichen Rechner gelöst werden. Ungeachtet dessen treten solche Probleme überaus häufig in der Praxis auf, z.B. bei vielen Planungsaufgaben, und es ist von großer ökonomischer Bedeutung, sie doch noch zu lösen, zumindest "so gut wie es geht". Die Vorlesung behandelt Methoden der Algorithmentheorie, die hierfür entwickelt wurden. Einige Stichpunkte: Approximationsalgorithmen, Branch-and-Bound, Parametrisierung. Es werden auch Grenzen dieser Methoden aufgezeichnet.		
Literatur: <ul style="list-style-type: none"> • Skript • Ausiello et al., Complexity and Approximation, Springer, Berlin, 1999. • J. Hromkovic, Algorithmics for Hard Problems, Springer, Berlin, 2001. 		
Modulteil: Algorithmen für NP-harte Probleme (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 2		

Prüfung

Algorithmen für NP-harte Probleme (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0052: Einführung in die Komplexitätstheorie <i>Introduction to Complexity Theory</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
Lernziele/Kompetenzen: Verständnis für zentrale Themen und Methoden der Komplexitätstheorie; die Fähigkeit, komplexitätstheoretische Sachverhalte und Fragen präzise zu formulieren und zu untersuchen.		
Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.		
Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)		
Voraussetzungen: Empfehlenswert: Gutes Verständnis des Stoffes aus Einführung in die Theoretische Informatik sowie Informatik III, insbesondere bzgl. Turing-Maschinen und Graphenalgorithmen. Modul Informatik 3 (INF-0111) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Einführung in die Komplexitätstheorie (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2		
Inhalte: Aufbauend auf den in den Grundvorlesungen Einführung in die Theoretische Informatik und Informatik III gelegten Grundlagen werden wichtige Aspekte der Komplexitätstheorie behandelt. Das Anliegen der Komplexitätstheorie ist es, die inhärente Schwierigkeit von Berechnungsproblemen zu untersuchen und somit die prinzipiellen Grenzen effizienter Algorithmen zu beleuchten.		
Literatur: <ul style="list-style-type: none"> • Skript • Christos H. Papadimitriou, Computational Complexity, Addison-Wesley, Reading, Mass., 1994. 		
Modulteil: Einführung in die Komplexitätstheorie (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 2		

Prüfung

Einführung in die Komplexitätstheorie (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0053: I/O-effiziente Algorithmen <i>I/O-Efficient Algorithms</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
Lernziele/Kompetenzen: Verständnis für den effizienten Umgang mit Speicherhierarchien; Kenntnis grundlegender I/O-effizienter Algorithmen, insbesondere für Sortieren und verwandte Probleme; Fähigkeit zur Analyse und Bewertung einfacher neuer Algorithmen im I/O-Modell; Verständnis für die Grenzen I/O-effizienter Algorithmen.		
Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.		
Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)		
Voraussetzungen: Empfehlenswert: Gutes Verständnis des Informatik III-Stoffes. Modul Informatik 3 (INF-0111) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: I/O-effiziente Algorithmen (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2		
Inhalte: Das klassische Berechnungsmodell der Random-Access-Machine (RAM) stößt zunehmend an seine Grenzen. Der Grund ist, dass moderne Rechner nicht über den "flachen" Speicher der RAM verfügen, bei dem alle Speicherzellen "gleichberechtigt" sind, sondern eine ausgefeilte Speicherhierarchie mit Caches, Hauptspeicher und Hintergrundspeicher(n) besitzen. Im Allgemeinen sind "näher am CPU" gelegene Speicher deutlich schneller, dafür aber kleiner, und ein effizienter Algorithmus muss versuchen, häufig benutzte Daten in Speicher mit kurzen Zugriffszeiten zu halten. In der Vorlesung werden wir uns, nach einer Einführung geeigneter Speichermodelle, aus theoretischer Sicht mit sogenannten I/O-effizienten oder "speicherbewussten" Algorithmen befassen, die die Anzahl der Datentransporte zwischen Stufen der Speicherhierarchie möglichst gering halten. Bereits für das Problem des Sortierens wird sich herausstellen, dass die "I/O-effiziente Welt" ganz anders aussieht als die "RAM-Welt".		
Literatur: <ul style="list-style-type: none"> • Skript • J.S. Vitter, Algorithms and data structures for external memory, Foundations and Trends in Theoretical Computer Science 2 (2008), pp. 305-474 		

Modulteil: I/O-effiziente Algorithmen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

I/O-effiziente Algorithmen (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Beschreibung:

Die Prüfung findet zum Ende des Semesters statt. Es besteht bei Bedarf die Möglichkeit einer Nachholprüfung im Folgesemester.

Modul INF-0054: Datenstrukturen <i>Data Structures</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
Lernziele/Kompetenzen: Kenntnis einer Reihe von nichtelementaren Datenstrukturen und ihrer Analyse; Fähigkeit zur Anpassung dieser Datenstrukturen an neue Anwendungen und zur Entwicklung neuer einfacher Datenstrukturen. Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.		
Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)		
Voraussetzungen: Empfehlenswert: Gutes Verständnis des Informatik III-Stoffes. Modul Informatik 3 (INF-0111) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	

Moduleile
Moduleil: Datenstrukturen (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 4
Inhalte: Datenstrukturen realisieren abstrakte Datentypen so, dass die Operationen der Datentypen besonders effizient ausgeführt werden können. Beispiele von Datenstrukturen sind balancierte Bäume und Hashtabellen. Datenstrukturen können mit objektorientierten Programmiersprachen als Klassen zur Verfügung gestellt werden. In der Vorlesung werden verschiedene Datenstrukturen behandelt, die über die in Informatik III behandelten Datenstrukturen hinausgehen, unter anderem die sogenannten dynamischen Bäume von Sleator und Tarjan, Range-Query-Strukturen und Suffix-Bäume.
Literatur: Skript
Zugeordnete Lehrveranstaltungen: Datenstrukturen (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i> Datenstrukturen realisieren abstrakte Datentypen so, dass die Operationen der Datentypen besonders effizient ausgeführt werden können. Beispiele von Datenstrukturen sind balancierte Bäume und Hashtabellen. Datenstrukturen können mit objektorientierten Programmiersprachen als Klassen zur Verfügung gestellt werden. In der Vorlesung werden verschiedene Datenstrukturen behandelt, die über die in Informatik III behandelten

Datenstrukturen hinausgehen, unter anderem die sogenannten dynamischen Bäume von Sleator und Tarjan, Range-Query-Strukturen und Suffix-Bäume.

Modulteil: Datenstrukturen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Datenstrukturen (Übung)

Veranstaltung wird online/digital abgehalten.

Bitte melden Sie sich in der Digicampus-Veranstaltung der Vorlesung an.

Prüfung

Datenstrukturen (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0056: Online-Algorithmen <i>Online Algorithms</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
Lernziele/Kompetenzen: Verständnis der Online-Problematik; Kenntnis fundamentaler Online-Probleme und -Algorithmen; Fähigkeit zum selbstständigen Entwurf einfacher Online-Algorithmen, zu ihrer kompetitiven Analyse mittels Potentialfunktionen und zu ihrer Bewertung.		
Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.		
Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Online-Algorithmen (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
Inhalte: Manchmal muss man Entscheidungen treffen, bevor alle relevanten Daten bekannt sind. Will man z. B. Aktien kaufen, so wäre es sehr hilfreich, über die künftige Entwicklung aller Aktienkurse informiert zu sein; aber es liegt in der Natur der Sache, dass man den Kauf tätigen muss, bevor diese Information vorliegt. Ein zweites Beispiel: Eine Funktaxizentrale muss nach jeder Bestellung einen der verfügbaren Wagen auswählen und zum Fahrgast schicken; mit Wissen über später eintreffende Anrufe könnten die Wagen vielleicht günstiger auf die Fahrgäste verteilt werden. Algorithmen, die Entscheidungen bei unvollständiger Information treffen, heißen Online-Algorithmen. Die Vorlesung behandelt Online-Algorithmen und ihre Analyse.		
Literatur: <ul style="list-style-type: none"> • Skript • A. Borodin und R. El-Yaniv, Online Computation and Competitive Analysis, Cambridge University Press, 1998. 		
Modulteil: Online-Algorithmen (Übung)		
Lehrformen: Übung		
Sprache: Deutsch		
SWS: 2		

Prüfung

Online-Algorithmen (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0130: Formale Methoden im Software Engineering <i>Formal Methods in Software Engineering</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
Lernziele/Kompetenzen: Fachbezogene Kompetenzen: Die Studierenden können mathematisch-formale Methoden für die Programmverifikation, speziell bei sicherheitskritischer Software einsetzen. Durch das Verständnis von Kalkülen trainieren sie die Fertigkeit zum logischen, konzeptuellen und analytischen Denken. Sie können Spezifikationen von Datenstrukturen entwerfen und deren Eigenschaften formal beweisen. Sie sind in der Lage, funktionale Eigenschaften von sequentiellen und nebenläufigen Programmen zu formulieren und dafür Beweise zu erstellen. Sie erwerben die Fertigkeit zur abstrakten, modularen Modellierung von Softwaresystemen und zur systematischen Entwicklung von korrekten Programmen aus diesen Modellen. Sie erwerben dadurch die Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen und können mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen abgeben. Sie erwerben Fähigkeiten, um Beiträge zur Wissenschaft zu leisten.		
Schlüsselqualifikationen: <ul style="list-style-type: none"> • Training des logischen Denkens • Analytisch-methodische Kompetenz • Erwerb von Abstraktionsfähigkeiten • Fähigkeiten zur Modellbildung für Softwaresysteme 		
Arbeitsaufwand: Gesamt: 240 Std. 60 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Formale Methoden im Software Engineering (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2		
Inhalte: Übergeordnetes Ziel ist die Produktion beweisbar korrekter Software. In der Vorlesung werden verschiedene klassische Methoden für die Programmverifikation im Kleinen behandelt. Darüber hinaus werde innovative Techniken für die formale Modellierung und Verifikation großer Systeme vermittelt. Als Werkzeug kommt das KIV-System zum Einsatz, das die formale Spezifikation und Verifikation von Systemen ermöglicht. Konkrete Inhalte sind: Algebraische Spezifikationen, interaktives Theorembeweisen, Hoare-Logik, Dynamische Logik, Temporallogik		

Literatur:

- Sperschneider, Antoniou: Logic: A Foundation for Computer Science, Addison Wesley 1991
- Loeckx, Ehrich, Wolf: Specification of Abstract Data Types, Wiley 1996
- Ausführliche Dokumentation
- Folienhandout

Modulteil: Formale Methoden im Software Engineering (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Prüfung

Formale Methoden im Software Engineering (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 45 Minuten

Modul INF-0156: Algebraische Beschreibung paralleler Prozesse <i>Algebraic Description of Concurrent Processes</i>		6 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen:</p> <p>Die Studierenden kennen die Prozessalgebra CCS und formale Beschreibungen verteilter Systeme in CCS. Sie kennen einen Mechanismus, mit dem man in CCS die operationale Semantik definiert, und können diesen Mechanismus auf konkrete Prozesse anwenden. Sie besitzen die Fähigkeit, verteilte Systeme entsprechend auf eine exakte, algebraische Weise zu modellieren, d.h. einen entsprechenden Prozess zu entwerfen.</p> <p>Sie wissen, welche Anforderungen man in diesem Kontext an Äquivalenzbegriffe stellen muss und lernen verschiedene solche Begriffe kennen. Sie können auf verschiedenen Weisen -- z.B. auch durch algebraische Gesetze -- formal prüfen, ob ein System eine, ebenfalls in CCS geschriebene, Spezifikation erfüllt.</p> <p>Sie können beurteilen, ob ein eine algebraische Gleichung ein gültiges Gesetz ist. Zur Vertiefung der Kenntnisse werden die meisten Resultate der Vorlesung bewiesen.</p> <p>Die Studierenden wissen, wie man CCS und die operationale Semantik in einer spezifischen Weise um Zeit erweitern kann, und sind dadurch in der Lage, auch andere Prozessalgebren mit anderen Semantik-Regeln zu verstehen und zu verwenden.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Methodik; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit zur Analyse und Strukturierung von Informatikproblemstellungen; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 180 Std.</p> <p>15 Std. Übung (Präsenzstudium)</p> <p>45 Std. Vorlesung (Präsenzstudium)</p> <p>75 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>22 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>23 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
<p>Voraussetzungen:</p> <p>Modul Einführung in die Theoretische Informatik (INF-0110) - empfohlen</p> <p>Modul Logik für Informatiker (INF-0155) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 5.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Algebraische Beschreibung paralleler Prozesse (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Deutsch</p> <p>SWS: 3</p>		
<p>Inhalte:</p> <p>Algebraische Spezifikation verteilter Systeme mittels der Prozessalgebra CCS; operationale Semantik mittels SOS-Regeln; Äquivalenz- bzw. Kongruenzbegriffe (starke und schwache Bisimulation, Beobachtungskongruenz); Nachweis von Kongruenzen mittels Axiomen; Einführung in eine Kombination von Bisimulation und Effizienzvergleich.</p> <p>Die meisten Resultate der Vorlesung werden bewiesen.</p>		

Literatur:

- R. Milner: Communication and Concurrency, Prentice Hall
- L. Aceto, A. Ingolfsdottir, K.G. Larsen, J. Srba: Reactive Systems. Cambridge University Press 2007
- J. Bergstra, A. Ponse, S. Smolka (eds.): Handbook of Process Algebras, Elsevier

Modulteil: Algebraische Beschreibung paralleler Prozesse (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 1

Prüfung

Algebraische Beschreibung paralleler Prozesse (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0157: Endliche Automaten <i>Finite Automata</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen: Nach der Teilnahme haben die Studierenden vertiefte Kenntnisse in der Automatentheorie und können sie verständlich erläutern. Sie können deterministische Automaten minimieren und das Verfahren mit guter Effizienz automatisieren. Sie können Schaltkreisverhalten und Mealy-Automaten ineinander übersetzen, und sie können mit geeigneten Ergebnissen reguläre von nicht-regulären Sprachen unterscheiden. Zur Vertiefung der Kenntnisse werden die meisten Resultate der Vorlesung bewiesen. Sie haben anspruchsvolle Modellierungen von Problemen mit endlichen Automaten kennengelernt und können sich in neue Anwendungen der Automatentheorie einarbeiten und ihre Korrektheit beurteilen.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Methodik; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Auswahl und sichere Anwendung geeigneter Methoden; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 20 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 37 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 48 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 45 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen: Modul Einführung in die Theoretische Informatik (INF-0110) - empfohlen Modul Informatik 3 (INF-0111) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 5.	Minimale Dauer des Moduls: 1 Semester
SWS: 3	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Modulteil: Endliche Automaten (Vorlesung mit integrierter Übung) Lehrformen: Vorlesung + Übung Sprache: Deutsch SWS: 3</p>		
<p>Inhalte: Die Vorlesung vertieft die Kenntnisse über Endliche Automaten aus der Grundvorlesung "Einführung in die theoretische Informatik", insbesondere über Minimierung und Abschlusseigenschaften. Sie behandelt eine Anwendung bei der Lösung diophantischer Gleichungen. Sie stellt Mealy-, Moore- und Büchi-Automaten vor und behandelt die Übersetzungen zwischen Mealy-Automaten und Schaltkreisverhalten. Die meisten Resultate der Vorlesung werden bewiesen.</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • Hopcroft, (Motwani, Ullman: Introduction to Automata Theory, Languages and Computation; deutsch: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie • Schöning: Theoretische Informatik kurz gefaßt. 5. Auflage • Thomas: Automata on Infinite Objects. Chapter 4 in Handbook of Theoretical Computer Science, Hrsg. van Leeuwen 		
<p>Zugeordnete Lehrveranstaltungen: Endliche Automaten (Vorlesung + Übung)</p>		

Veranstaltung wird online/digital abgehalten.

Diese Vorlesung befasst sich mit endlichen Automaten in Ergänzung der Grundvorlesung "Einführung in die theoretische Informatik". Insbesondere werden die Minimierung endlicher Automaten, Automaten mit Ausgabe sowie Automaten für unendliche Wörter behandelt. Um die Nützlichkeit endlicher Automaten aufzuzeigen, wird z.B. kurz eine Verbindung zur Logik hergestellt.

Prüfung

Endliche Automaten (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0161: Petrinetze - eine Theorie paralleler Systeme <i>Petri Nets - A Theory of Concurrent Systems</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen: Die Studierenden kennen verschiedene Varianten von Petrinetzen und entsprechende formale, Graphen-basierte Beschreibungen paralleler bzw. nebenläufiger Systeme. Anhand verschiedener Verhaltensbegriffe (Schalt- und Schrittfolgen, Sprache, Failure-Semantik) lernen sie die neuartigen Aspekte der Abläufe solcher Systeme kennen. Sie werden befähigt, wichtige Systemeigenschaften mit Petrinetz-spezifischen Methoden zu beweisen oder zu widerlegen. Auf dieser Grundlage können sie parallele bzw. nebenläufige Systeme mit Petrinetzen formal modellieren sowie analysieren, unter welchen Aspekten sich verschiedene Modelle gleichen oder unterscheiden. Sie lernen einige Entscheidbarkeitsprobleme kennen und können eigene Beweise dazu führen. Zur Vertiefung der Kenntnisse werden die meisten Resultate der Vorlesung bewiesen.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Methodik; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fertigkeit zur Analyse und Strukturierung von Informatikproblemstellungen; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Modul Einführung in die Theoretische Informatik (INF-0110) - empfohlen		
Angebotshäufigkeit: unregelmäßig (i. d. R. im WS)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Petrinetze - eine Theorie paralleler Systeme (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Graphenbasierte Modellierung paralleler Systeme mittels verschiedener Varianten von Petrinetzen; verschiedene Verhaltensbeschreibungen (Schalt- und Schrittfolgen, Sprache, Failure-Semantik); Begriffe und Techniken der Verhaltensanalyse (Verklemmung, Lebendigkeit, Fairness; S- und T-Invarianten, Überdeckbarkeitsgraph); Entscheidbarkeitsprobleme. Die meisten Resultate der Vorlesung werden bewiesen.</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • Desel, Reisig, Rozenberg (eds.): Lectures on Concurrency and Petri Nets. Advances in Petri Nets. Springer, LNCS 3098 • Peterson: Petri Net Theory and the Modelling of Systems. Prentice Hall • Reisig: Petrinetze - Eine Einführung. 2. Auflage; Springer 		

Modulteil: Petrinetze - eine Theorie paralleler Systeme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Petrinetze - eine Theorie paralleler Systeme (Mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0163: Verteilte Algorithmen <i>Distributed Algorithms</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen: Die Studierenden lernen formale Modelle zur Beschreibung verteilter Algorithmen in synchronen und asynchronen Netzwerke sowie die entsprechenden Methoden zum Nachweis der Korrektheit und des Aufwands kennen. Sie können beschreiben, wie Fehler in den beiden Modellen behandelt werden. Die Studierenden lernen wichtige Algorithmen und ihren Aufwand kennen, können sie in konkreten Netzwerken simulieren und verstehen ihre Korrektheit. Sie werden befähigt, solche Algorithmen zu modifizieren und zu analysieren, so dass sie zugehörige Korrektheitsbeweise und Aufwandsbestimmungen prüfen oder selbst entwickeln können. Sie erlangen damit ein vertieftes Verständnis für die Probleme und Problemlösungen in verteilten Systemen.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Methodik; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fertigkeit zur Analyse und Strukturierung von Informatikproblemstellungen; Kenntnisse der Vor-/Nachteile von Entwurfsalternativen, Bewertung im jeweiligen Anwendungszusammenhang; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen: Modul Einführung in die Theoretische Informatik (INF-0110) - empfohlen Modul Informatik 3 (INF-0111) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Verteilte Algorithmen (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 4</p>		
<p>Inhalte: Algorithmen für Grundprobleme in Netzwerken wie Zugriff auf gemeinsame Ressourcen, Aufbau geeigneter Kommunikationsstrukturen und Konsens; es werden synchrone und asynchrone Netzwerke und Fehlertoleranz betrachtet, der Aufwand bestimmt und nachgewiesen sowie Korrektheitsbeweise geführt.</p>		
<p>Literatur: Nancy Lynch, Distributed Algorithms. Morgan Kaufmann 1996</p>		
<p>Zugeordnete Lehrveranstaltungen: Verteilte Algorithmen (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i> Diese Vorlesung präsentiert Algorithmen für Grundprobleme in Netzwerken mit zahlreichen Prozessoren, wie sie in den Bereichen Telekommunikation, Prozeßkontrolle und verteilte Informationssysteme auftreten, und untersucht</p>		

die Effizienz dieser Algorithmen. Grundprobleme sind der Zugriff auf gemeinsame Ressourcen (Problem des gegenseitigen Ausschlusses), Kommunikation (Aufbau geeigneter Strukturen zur Verbreitung von Daten im Netzwerk) und Konsens (z.B. ob eine Transaktion in einem verteilten Datenbanksystem abgebrochen werden muß). Dabei sollen Lösungen fehlertolerant sein, also auch bei fehlerhaftem Verhalten oder Ausfall einiger Prozessoren oder Kommunikationsverbindungen korrekt arbeiten. In der Vorlesung wird die Korrektheit der behandelten Algorithmen in der Regel bewiesen, da sie in vielen Anwendungen (in Telefon-, Reservierungs-, Bank- oder Luftfahrt-Kontroll-Systemen) essentiell ist.
... (weiter siehe Digicampus)

Modulteil: Verteilte Algorithmen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Verteilte Algorithmen (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Verteilte Algorithmen (Mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0201: Platzeffiziente Algorithmen <i>Space-Efficient Algorithms</i>		5 ECTS/LP
Version 1.1.0 (seit WS15/16) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
<p>Lernziele/Kompetenzen: Die Studierenden sind in der Lage, platzeffiziente Algorithmen zu verstehen, zu analysieren und selbst zu entwerfen. Sie verstehen die häufig notwendige Abwägung zwischen Zeit und Platz und kennen wichtige Entwurfsmethoden und grundlegende Datenstrukturen für platzeffiziente Algorithmen ebenso wie eine Anzahl konkreter platzeffizienter Algorithmen.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
<p>Voraussetzungen: Teilnehmer der Vorlesung sollten über gute algorithmische Kenntnisse verfügen, insbesondere im Bereich Graphenalgorithmen. Modul Informatik 3 (INF-0111) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Platzeffiziente Algorithmen (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Manchmal hat ein Algorithmus eine große Eingabe, aber nur wenig frei beschreibbaren Arbeitsspeicher. Zum Beispiel könnte die Eingabe im Internet für Anfragen zur Verfügung stehen, aber in ihrer Gesamtheit so riesig sein, dass es unmöglich oder unpraktisch ist, sie auf den lokalen Rechner herunterzuladen. Die Vorlesung beschäftigt sich aus theoretischer Sicht mit Algorithmen, die mit weniger Arbeitsspeicher als klassische Algorithmen für dieselben Probleme auskommen. Der Fokus liegt auf Graphenproblemen wie die Durchführung einer Tiefensuche oder die Berechnung kürzester Wege, aber auch Sortieren und platzeffiziente Datenstrukturen kommen zur Sprache. Ein Großteil der in der Vorlesung vorgestellten Ergebnisse wurde seit 2014 am Lehrstuhl für Theoretische Informatik erzielt. Die Vorlesung behandelt somit ein sehr aktives und aktuelles Forschungsgebiet.</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • Skript 		

Modulteil: Platzeffiziente Algorithmen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Platzeffiziente Algorithmen (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Beschreibung:

Die Prüfung findet zum Ende des Semesters statt. Es besteht bei Bedarf die Möglichkeit einer Nachholprüfung im Folgesemester.

Modul INF-0243: Process Mining <i>Process Mining</i>		6 ECTS/LP
Version 1.1.0 (seit SoSe17) Modulverantwortliche/r: Prof. Dr. Robert Lorenz		
<p>Lernziele/Kompetenzen: Teilnehmer verstehen die folgenden wesentlichen Konzepte der Informatik auf einem vertieften wissenschaftlichen Niveau mit ihren mathematisch-formalen Grundlagen: Ereignis-Log, Halbordnung und partielle Sprache, Petrinetz, Nebenläufigkeit, sequentielle und kausale Semantik eines nebenläufigen Systems, Synthese von nebenläufigen Systemen, Geschäftsprozess, Process Mining, Process Discovery. Sie können nebenläufige Systeme mittlerer Komplexität in einer geeigneten Modellierungssprache modellieren, sowie zu einem gegebenen nebenläufigen System verschiedene Verhaltensmodelle zur Analyse und Simulation generieren, bewerten und ineinander überführen. Sie kennen die verschiedenen Process Mining Anwendungsfälle mit dazu passenden Lösungstechniken und Qualitätskriterien zu deren Bewertung. Teilnehmer verfügen über eine fortgeschrittene mathematisch formale Methodik für die Analyse und Formalisierung komplexer Process Discovery Probleme. Sie sind in der Lage, geeignete Algorithmen zu deren Lösung bzgl. qualitativer und quantitativer Kriterien sicher begründet auszuwählen, sowie diese anzuwenden und die Ergebnisse zu interpretieren und zu bewerten.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen, konzeptionellen und algorithmischen Denken; Eigenständiges Arbeiten mit Lehrbüchern und englischsprachiger Fachliteratur; Verständliche Präsentation von Ergebnissen; Qualitätsbewußtsein; Akribie; Grundsätze guter wissenschaftlicher Praxis; Wissenschaftliche Methodik;</p>		
<p>Arbeitsaufwand: Gesamt: 180 Std. 45 Std. Vorlesung (Präsenzstudium) 15 Std. Übung (Präsenzstudium) 23 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 75 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 22 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Process Mining (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch / Englisch SWS: 3</p>		
<p>Inhalte: Die Veranstaltung gibt einen fundierten Überblick über das Forschungsgebiet Process Mining:</p> <ul style="list-style-type: none"> - Modelling Techniques: Workflow Nets, Partial Languages - Discovering algorithms - Conformance Checking - Process Enhancement - The PROM Framework 		

Literatur:

Wil M. P. van der Aalst:

Process Mining - Data Science in Action, Second Edition. Springer 2016, ISBN 978-3-662-49850-7

Wil M. P. van der Aalst, Boudewijn F. van Dongen: Discovering Petri Nets from Event Logs. Trans. Petri Nets and Other Models of Concurrency 7: 372-422 (2013)

Robin Bergenthum, Jörg Desel, Robert Lorenz, Sebastian Mauser: Process Mining Based on Regions of Languages. BPM 2007: 375-383

Modulteil: Process Mining (Übung)

Lehrformen: Übung

Sprache: Deutsch / Englisch

SWS: 1

Prüfung

Process Mining (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0310: Perlen der Algorithmik <i>Algorithmic Gems</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an der Vorlesung haben die Studierenden einen guten Überblick über mehrere Teilgebiete der Algorithmik. Sie kennen verschiedene Berechnungsmodelle und verschiedene Arten der Bewertung und der Analyse von Datenstrukturen und Algorithmen und können diese auf neue Problemstellungen anwenden. Sie sind vertraut mit Algorithmen und Datenstrukturen, die nicht zum Standardkanon eines Informatikstudiums gehören.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen: Teilnehmer der Vorlesung sollten über gute algorithmische Kenntnisse verfügen. Modul Informatik 3 (INF-0111) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Moduleil: Perlen der Algorithmik (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
Inhalte: Die Vorlesung stellt eine Reihe meist unabhängiger Algorithmen und Datenstrukturen vor und versucht so, den Teilnehmern einen Überblick über Sichtweisen, Ziele, Methoden und Ergebnisse vieler Teilgebiete der Algorithmik zu vermitteln. Die konkrete Themenauswahl basiert auf Kriterien wie theoretische oder praktische Wichtigkeit, relative Einfachheit, Eleganz und Neuheit.		
Literatur: Ein englischsprachiges Skript wird zur Verfügung gestellt.		
Moduleil: Perlen der Algorithmik (Übung)		
Lehrformen: Übung		
Sprache: Deutsch		
SWS: 2		

Prüfung

Perlen der Algorithmik

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0371: Approximation Algorithms <i>Approximation Algorithms</i>		5 ECTS/LP
Version 1.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Tobias Mömke		
<p>Lernziele/Kompetenzen: Developing an understanding of central topics in the field of approximation algorithms; acquiring powerful mathematical tools to analyze algorithms; improve the ability to abstract and systematically solve optimization problems.</p> <p>Schlüsselqualifikationen: Ability to build intuitive understanding of mathematical formalisms; ability to identify core properties of optimization problems; deep understanding of powerful mathematical tools</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
<p>Voraussetzungen: Basic knowledge of Algorithms and Data Structures (e.g., "INF-0111: Informatik 3") and Theoretical Computer Science (e.g., "INF-0110: Einführung in die Theoretische Informatik").</p>		
<p>Angebotshäufigkeit: unregelmäßig</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 4</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
Modulteile		
<p>Modulteil: Approximation Algorithms (Vorlesung) Lehrformen: Vorlesung Sprache: Englisch SWS: 2</p>		
<p>Inhalte: Given an NP-hard optimization problem, how well can it be approximated in polynomial time? It is exciting and challenging to understand the approximability of fundamental optimization problems. This course mainly focuses on upper bounds, i.e., designing efficient approximation algorithms. In this course, we will study several classes of problems, such as packing problems, network design, and graph problems. We will cover central algorithmic techniques for designing approximation algorithms, including greedy algorithms, dynamic programming, linear and semi-definite programming, and randomization.</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • David P. Williamson and David B. Shmoys, The Design of Approximation Algorithms, Cambridge University Press. • Vijay V. Vazirani, Approximation Algorithms, Springer. 		
<p>Zugeordnete Lehrveranstaltungen: Approximation Algorithms (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i> Given an NP-hard optimization problem, how well can it be approximated in polynomial time? It is exciting and challenging to understand the approximability of fundamental optimization problems. This course mainly focuses</p>		

on upper bounds, i.e., designing efficient approximation algorithms. In this course, we will study several classes of problems, such as packing problems, network design, and graph problems. We will cover central algorithmic techniques for designing approximation algorithms, including greedy algorithms, dynamic programming, linear and semi-definite programming, and randomization. This course does not require specific prerequisite, other than basic knowledge in algorithms and in data structures.

Modulteil: Approximation Algorithms (Übung)

Lehrformen: Übung

Sprache: Englisch / Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Approximation Algorithms (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Approximation Algorithms

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0037: Praktikum Automotive Software Engineering <i>Practical Module Automotive Software Engineering</i>		6 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
Lernziele/Kompetenzen: Nach der Teilnahme am Praktikum Automotive Software Engineering verstehen die Studierenden praxisnaher Problemstellungen höherer Komplexität im Bereich der Entwicklung und Absicherung von Fahrerassistenzsystemen mit aktuellen Methoden und Tools der modellbasierten Entwicklung eingebetteter Systeme. Die Studierenden erlangen tiefergehende fachspezifische als auch fächerübergreifende Kenntnisse und Fähigkeiten, beispielsweise aus der Regelungstechnik, Fahrphysik und Mathematik. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren, zu präsentieren und verständlich zu dokumentieren.		
Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Projektmanagementfähigkeiten		
Arbeitsaufwand: Gesamt: 180 Std. 90 Std. Praktikum (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)		
Voraussetzungen: Empfohlen wird die Teilnahme an einem der beiden links aufgeführten Seminare. Modul Seminar Grundlagen des Software Engineering für Automotive Systems (BA) (INF-0027) - empfohlen Modul Seminar Grundlagen des Software Engineering für Automotive Systems (MA) (INF-0040) - empfohlen		
Angebotshäufigkeit: unregelmäßig (i. d. R. im WS)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Automotive Software Engineering (MA) Lehrformen: Praktikum Sprache: Deutsch Angebotshäufigkeit: jedes Wintersemester SWS: 6		

Inhalte:

Im Automotive-Praktikum lernen die Teilnehmer wie verschiedene ausgewählte Funktionen innerhalb von Fahrzeugen simuliert und analysiert werden können. In einem zweitägigen Einführungskurs werden die benötigten theoretischen Grundlagen für die Bearbeitung der Praxisaufgabe gelegt. Während des Einführungskurses wird das Modell eines Antiblockiersystems (ABS) auf realen Steuergeräten behandelt. Die Teilnehmer lernen dabei u.a. die im Automotive-Umfeld häufig eingesetzte Modellierungswerkzeug-Kombination „Matlab/Simulink“ sowie das graphische Simulations- und Analyse-Tool „CarMaker“ kennen und erhalten einen praktischen Einblick in die Funktionsweise von FlexRay-Bussystemen.

Nach dem Einführungskurs soll in Gruppen von zwei bis drei Teilnehmern das Modell eines ACC-Systems (Adaptive Cruise Control) erstellt, simuliert und verifiziert werden.

Literatur:

abhängig vom Thema

Prüfung

Praktikum Automotive Software Engineering (MA)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0066: Organic Computing II <i>Organic Computing II</i>		5 ECTS/LP
Version 1.3.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen: Teilnehmer sind in der Lage, wesentliche Konzepte und Methoden des Organic Computing (OC) wiederzugeben. Dazu zählen unter anderem: Selbst-Organisation, Selbst-Adaption, Robustheit, Flexibilität, Observer/Controller-Architekturen (O/C), Selbst-X Eigenschaften, Extended Classifier Systems (XCS), Genetische Algorithmen (GA), Partikel Swarm Optimization, Influence Detection und Trust. Sie können zudem begründen, wieso es sinnvoll ist OC-Systeme zu betrachten und praxisrelevante Beispiele nennen, in welchen OC-Techniken angewendet werden sollten. Die Teilnehmer sind befähigt, größere Softwaresysteme mit Hilfe der O/C-Architektur zu entwickeln, sowie diese mit passenden OC-Techniken zu befüllen und sich, falls erforderlich, neue OC-Techniken anzueignen ein gegebenes Problem zu lösen. Die Studenten haben die Fähigkeit zum Lösen von praktischen Aufgaben mit Hilfe des XCS, dem Swarming und einem GA. Sie können zudem ein Konzept für ein OC-System ausarbeiten und beurteilen, sowie wissenschaftliche Bewertungen im Bezug auf OC-Systeme durchführen.</p> <p>Schlüsselqualifikationen: Fähigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen, Fähigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten., Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen, Fertigkeit der Zusammenarbeit in Teams.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile
<p>Modulteil: Organic Computing II (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>
<p>Inhalte: Die Vorlesung "Organic Computing" vermittelt Ansätze zur Organisation von komplexen vernetzten Systemen, die aus einer Vielzahl von autonomen Teilsystemen bestehen. Dazu werden zunächst Anforderungen und Ziele solcher Systeme definiert und diskutiert. Darüber hinaus werden Konzepte aus dem Bereich der Systemarchitekturen und Ansätze aus dem Bereich naturanaloger Algorithmen dargestellt und bewertet. In allen Teilen werden Bezüge zu konkreten Anwendungsgebieten gegeben. Die zugehörige Übung bietet die Möglichkeit, die erlernten Ansätze zu vertiefen und beispielhaft anzuwenden.</p>

Literatur:

- Folien
- Müller-Schloer und Tomforde: Organic Computing - Technical Systems for Survival in the Real World, Springer International Publishing, 2018, ISBN 978-3-319-68477-2
- Müller-Schloer et al.: Organic Computing - A Paradigm Shift for Complex Systems, Birkhäuser Verlag, Basel, 2011, ISBN 978-3034801294
- Würtz (ed.): Organic Computing (Understanding Complex Systems), SpringerVerlag Berlin, 2008, ISBN 978-3540776567
- Mitchell: Machine Learning, The McGraw-Hill Companies, 1997, ISBN 978-0071154673
- Goldberg: Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, 1989, ISBN 978-0201157673
- Michalewicz, Fogel: How to Solve it: Modern Heuristics, Springer Verlag Berlin, 2004, ISBN 978-3540224945
- Tomforde: Runtime Adaptation of Technical Systems, Südwestdeutscher Verlag für Hochschulschriften, 2012, ISBN 978-3838131337

Modulteil: Organic Computing II (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Organic Computing II (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Beschreibung:

Die Prüfung kann jedes Semester zu Beginn und Ende der vorlesungsfreien Zeit abgelegt werden.

Modul INF-0077: Suchmaschinen <i>Search Engines</i>		8 ECTS/LP
Version 1.2.0 (seit SoSe14) Modulverantwortliche/r: PD Dr. Markus Endres		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an der Veranstaltung sind die Studierenden in der Lage die Konzepte und Methoden, Verfahren, Techniken und Technologien von Suchmaschinen zu verstehen und zu bewerten. Hierfür sind vertiefende Kenntnisse im Bereich der Suchmaschinentechologie notwendig. Darunter zählen z.B. die Architektur einer Suchmaschine, Web Crawler, Textprocessing, Indexierung, verschiedene Retrieval Modelle, die Evaluation von Suchalgorithmen, Clusterverfahren, Link Analyse, uvm.</p> <p>Die Studierenden erwerben somit Kenntnisse verschiedener Suchmaschinenaspekte und können die Vor- und Nachteile in praxisrelevanten Aufgabenstellungen abwägen und beurteilen. Zudem können die Studierenden nach der Veranstaltung wissenschaftlich aussagekräftige Bewertungen erstellen und diese in praxisrelevante Lösungsstrategien umsetzen. In der Veranstaltung werden zudem relevante Themen aus der Forschung anhand aktueller Veröffentlichungen angesprochen, so dass die Studierenden die Fähigkeit erwerben, ihr vorhandenes Wissen selbständig zu erweitern.</p>		
<p>Schlüsselqualifikationen: Fachspezifische Vertiefung; Umsetzen fachlicher Lösungskonzepte in Programme und Modelle; Kenntnisse der Vor-/Nachteile von Entwurfsalternativen und Bewertung im jeweiligen Anwendungszusammenhang; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit, mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen anzustellen; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen;</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Modul Datenbanksysteme (INF-0073) - empfohlen		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Suchmaschinen (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 4		
Inhalte: Die Vorlesung behandelt grundlegende Konzepte von Suchmaschinen, Volltext-Suche, SQL-Suchmaschinen und Präferenz-Suchmaschinen (Preference SQL) sowie deren Implementierung.		

Literatur:

- M. Levene: An Introduction to Search Engines and Web Navigation
- R. Baeza-Yates, B. Ribeiro-Neto: Modern Information Retrieval
- I. H. Witten, M. Gori, T. Numerico: Web Dragons
- W. Kießling: Foundations of Preferences in Database Systems
- W. Kießling: Preference Queries with SV-Semantics

Modulteil: Suchmaschinen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Suchmaschinen (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0078: Datenbankprogrammierung (Oracle) <i>Database Programming (Oracle)</i>		5 ECTS/LP
Version 1.2.0 (seit SoSe14) Modulverantwortliche/r: PD Dr. Markus Endres		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an der Veranstaltung sind die Studierenden in der Lage vertiefte Oracle-Kenntnisse in praxisrelevanten Aufgaben anzuwenden. Hierzu ist erweitertes Wissen im Bereich DB-System-Architektur, DB-Entwurfstheorie, Administration, erweitertes SQL, PL/SQL-Programmierung, Baumstrukturen, XML und OLAP notwendig. Die Studierenden erlangen eine fachspezifische Vertiefung im Bereich der Datenbanksysteme und somit die Fertigkeit zur Analyse und Strukturierung von Problemstellungen. Darüber hinaus können die Studierenden diese komplexen, praxisrelevanten Aufgabenstellungen auf dem Gebiet Datenbanken, insbesondere unter Verwendung von Oracle, analysieren, bewerten und lösen. Sie kennen die Vor- und Nachteile unterschiedlicher ER-Modellierungen und können durch logisches und konzeptionelles Denken eine geeignete Lösung für komplexe Problemstellungen unter praxisnahen Randbedingungen verfassen.</p> <p>Schlüsselqualifikationen: Umsetzen fachlicher Lösungskonzepte in Programm eund Modelle, Fachspezifische Vertiefung; Fachübergreifende Kenntnisse; Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen; Kenntnisse der Vor-/ Nachteile von Entwurfalternativen und Bewertung im jeweiligen Zusammenhang; Auswahl und sichere Anwendung geeigneter Methoden; Kenntnisse von praxisrelevanten Aufgabenstellungen; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen;</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Modul Datenbanksysteme (INF-0073) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Datenbankprogrammierung (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch / Englisch		
SWS: 2		
Inhalte: Die Vorlesung behandelt Problemlösungsstrategien unter Zuhilfenahme einer Oracle-Datenbank. Dazu werden die Oracle-Architektur, Zugriffsrechte, Transformation von ER nach SQL, Oracle SQL, Aktive Inhalte wie PL/SQL und Java in Oracle, XML-Unterstützung in Oracle, Baumstrukturen, Tuning, Backup und Recovery behandelt.		

Literatur:

- R. Elmasri, S. Navathe: Fundamentals of Database Systems
- S. Melton: Understanding the New SQL: A Complete Guide
- Oracle 11g Online-Dokumentation

Zugeordnete Lehrveranstaltungen:

Datenbankprogrammierung (Oracle) (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Datenbanken haben sich als allgegenwärtiges Werkzeug im öffentlichen, wissenschaftlichen und wirtschaftlichen Leben etabliert. Diese Vorlesung wendet sich an DB-Interessierte, die ihre vorhandenen Kenntnisse aus einer grundlegenden Datenbankvorlesung mit Hilfe von Oracle vertiefen bzw. erweitern wollen. Daher ist die Vorlesung insbesondere für Studierende geeignet, die ihren Schwerpunkt im Bereich Datenbanken- und Informationssysteme setzen bzw. vertiefte praktische Kenntnisse erwerben wollen.

Modulteil: Datenbankprogrammierung (Übung)

Lehrformen: Übung

Sprache: Deutsch / Englisch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Datenbankprogrammierung (Oracle) (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Datenbankprogrammierung (Klausur)

Klausur / Prüfungsdauer: 60 Minuten

Modul INF-0088: Bayesian Networks <i>Bayesian Networks</i>		5 ECTS/LP
Version 1.0.0 (seit WS15/16) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Bayessche Netze gehören zu den vielseitigsten statistischen Methoden des maschinellen Lernens. Nach einer erfolgreichen Teilnahme an diesem Modul verstehen und vertiefen Teilnehmer die Kernprinzipien von Bayesschen Netzen und können diese auf viele praxisrelevante Probleme in unterschiedlichen Fachgebieten anwenden. Diese umfassen unter anderem die Robotik, Websuche, intelligente Agenten, automatisierte Diagnosesysteme und medizinische Systeme. Die Studierenden sind in der Lage, Bayessche Netze zu verstehen, anzuwenden und fachübergreifende Problemstellungen in diesem Kontext zu analysieren und zu evaluieren. Die Teilnahme an diesem Modul fördert Fertigkeiten zum logischen, analytischen und konzeptionellen Denken. Studierende sind in der Lage, mittels Bayesscher Netze wissenschaftlich aussagekräftige Bewertungen zu erstellen.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Logik; umsetzen fachlicher Lösungskonzepte in Modelle; fachübergreifende Kenntnisse; Entwicklung und Umsetzung von Lösungsstrategien komplexer Problemstellungen; systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen</p>		
<p>Bemerkung: Die gleichzeitige Einbringung von diesem Modul und INF-0263 ist nicht möglich.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen: keine</p>		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 5.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Bayesian Networks (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte:</p> <ol style="list-style-type: none"> 1. Basics of Probability Theory 2. Example: Bayesian Network based Face Detection 3. Inference 4. Influence Diagrams 5. Parameter Learning 6. Example: probabilistic Latent Semantic Analysis (pLSA) 		

Literatur:

- Richard E. Neapolitan. Learning Bayesian Networks. Prentice Hall Series in Artificial Intelligence, 2004. ISBN 0-13-012534-2
- Daphne Koller, Nir Friedman. Probabilistic Graphical Models: Principles and Techniques. The MIT Press, 2009. 978-0262013192

Modulteil: Bayesian Networks (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Bayesian Networks (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Beschreibung:

Die Prüfung kann jedes Semester in der Prüfungszeit abgelegt werden.

Modul INF-0093: Probabilistic Robotics <i>Probabilistic Robotics</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach einer erfolgreichen Teilnahme an diesem Modul verstehen Teilnehmer Grundlagen und vertiefende Fragestellungen und Algorithmen der Robotik (z.B. rekursive Zustandsschätzung, gaußsche- und nicht-parametrische Filter, Kalman Filter, Bewegung und Lokalisierung, Perzeption, Kartierung, SLAM) aus wahrscheinlichkeitstheoretischer Sicht und können erlernte Konzepte auf komplexe, praxisrelevante Aufgabenstellung anwenden. Die Studierenden sind in der Lage, Problemstellungen in diesem Kontext zu analysieren und zu evaluieren. Die Teilnahme an diesem Modul fördert Fertigkeiten zum logischen, analytischen und konzeptionellen Denken auf dem Gebiet der wahrscheinlichkeitstheoretischen Robotik. Studierende können aus den erlernten Konzepten zielgerichtet geeignete Methoden auswählen, sicher anwenden und auf neue, auch fachfremde, Fragestellungen übertragen. Das Modul vermittelt Kompetenzen zum Erkennen aktueller Forschung und bedeutenden technischen Entwicklungen auf diesem Gebiet.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Logik; umsetzen fachlicher Lösungskonzepte in Modelle; fachübergreifende Kenntnisse; Entwicklung und Umsetzung von Lösungsstrategien komplexer Problemstellungen; systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Probabilistic Robotics (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch Angebotshäufigkeit: jedes Wintersemester SWS: 2</p>		

Inhalte:

1. Introduction to Probabilistic Robotics
2. Recursive State Estimation
3. Gaussian Filters
4. Nonparametric Filters
5. Robot Motion
6. Robot Perception
7. Mobile Robot Localization: Markow and Gaussian
8. Mobile Robot Localization: Grid and MonteCarlo
9. Occupancy Grid Mapping
10. SLAM

Literatur:

Sebastian Thrun, Wolfram Burgard, Dieter Fox. Probabilistic Robotics. Springer Verlag.

Modulteil: Probabilistic Robotics (Übung)

Lehrformen: Übung

Sprache: Deutsch

Angebotshäufigkeit: jedes Wintersemester

SWS: 2

Prüfung

Probabilistic Robotics (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Beschreibung:

Die Prüfung kann jedes Semester in der Prüfungszeit abgelegt werden.

Modul INF-0129: Softwaretechnik II <i>Software Engineering II</i>		8 ECTS/LP
Version 2.0.0 (seit WS16/17) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen:</p> <p>Fachbezogene Kompetenzen:</p> <p>Nach dem erfolgreichen Besuch des Moduls sind die Teilnehmer in der Lage, fortgeschrittene Verfahren der agilen Softwareentwicklung, des Requirements Engineerings, des Testens und des Refactorings von Softwaresystemen anzuwenden und miteinander zu verbinden. Sie können die Anwendbarkeit der verschiedenen Verfahren und Methoden situationsspezifisch beurteilen.</p> <p>Die Teilnehmer verstehen die Aufgaben des Requirements Engineering und sind in der Lage, wesentliche Methoden der Requirements-Erfassung und Dokumentation (Use-Cases, Satzschablonen, formale Modelle) anzuwenden und die Eignung der verschiedenen Dokumentationsformen im Projektkontext zu bewerten. Sie haben die Fertigkeit zum analytischen und konzeptionellen Denken und können so systematisch komplexe Kundenanforderungen analysieren und können diese formulieren und in geeigneten Modellen darstellen.</p> <p>Die Studierenden verstehen einen agilen Entwicklungsprozess und die Basisprinzipien agiler Vorgehensweisen. Sie können den agilen Entwicklungsprozess auf Projekte übertragen und seine Anwendbarkeit im Projektkontext beurteilen. Sie kennen agile Praktiken und deren Wechselwirkungen und können die Praktiken einsetzen. Sie verstehen die Aufgaben der beteiligten Rollen und die Bausteine des Entwicklungsprozesses.</p> <p>Die Studierenden verstehen die Aufgaben des Testens und die Teilschritte des Testprozesses. Sie kennen Verfahren zur Bestimmung von Testfällen und zur Bewertung der Adäquatheit von Testsuiten und können diese anwenden.</p> <p>Die Studierenden kennen die Prinzipien guten objektorientierten Designs und können durch Refactoring die Qualität des Designs eines Systems verbessern. Sie können Ideen und Konzepte sicher und überzeugend darstellen und haben die Fähigkeit zur Zusammenarbeit im Team.</p> <p>Die Studierenden erwerben insgesamt Kenntnisse in Methoden zur Entwicklung großer Softwaresysteme und der Konstruktion von Abstraktionen sowie die Fähigkeit zu deren Anwendung im Projektkontext. Sie können alternative Vorgehensweisen wissenschaftlich fundiert bewerten und für ein Projekt geeignete auswählen.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Abstraktionsfähigkeit • Moderieren fachlicher Sitzungen • Erlernen des eigenständigen Arbeitens mit Lehrbüchern und Arbeit in selbstorganisierten Teams • Kenntnisse von praxisrelevanten Aufgabenstellungen 		
<p>Arbeitsaufwand:</p> <p>Gesamt: 240 Std.</p> <p>90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>30 Std. Übung (Präsenzstudium)</p> <p>60 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen:		
Programmierkenntnisse in Java (empfohlen) Modul Softwaretechnik (INF-0120) - empfohlen		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester

SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
------------------	---	--

Moduleile**Modulteil: Softwaretechnik II (Vorlesung)****Lehrformen:** Vorlesung**Sprache:** Deutsch**SWS:** 2**Inhalte:****Agile Softwareentwicklung:**

- Entwicklungsmethoden (Scrum)
- Agile Praktiken
- Agile Werte, Prinzipien und Methoden

Refactoring

- Code Smells
- Prinzipien des objektorientierten Designs
- Wichtige Refactorings

Testen

- Testprozess und Ziele des Testens
- Testarten
- Methoden zur Testfallgewinnung
- Adäquatheitskriterien beim Blackbox- und Whitebox-Testen

Requirements Engineering

- Aufgaben, Begriffe und Artefakte
- RE-Prozess
- Techniken zur Requirements-Elicitation, -Analyse und -Dokumentation
- Qualitätskriterien für Software-Requirements

Literatur:

- Pohl, Rupp: Basiswissen Requirements Engineering, dpunkt Verlag 2009
- U. Hammerschall, G. Benekean: Software Requirements, Pearson 2013
- S. Robertson, J. Robertson: Mastering the Requirements Process, Addison-Wesley 2013
- Bleek, Wolf: Agile Softwareentwicklung, dpunkt Verlag 2008
- R. Pichler: Scrum, dpunkt.verlag 2008
- Spillner, Linz: Basiswissen Softwaretest, dpunkt Verlag 2005
- Fowler: Refactoring, Addison-Wesley 1999
- Vorlesungsfolien mit schriftlichen Ergänzungen und Anmerkungen

Zugeordnete Lehrveranstaltungen:**Softwaretechnik 2 (Vorlesung)**

Veranstaltung wird als Hybrid/gemischt abgehalten.

Die Veranstaltung findet im Wesentlichen digital statt. Vorlesungsinhalte werden als Screencast zum Download angeboten. Zusätzlich gibt es pro Woche höchstens einen synchronen Termin, bei dem Fragen zum Stoff gestellt werden können und die Übungen behandelt werden. Für den synchronen Termin ist im Stundenplan Dienstag ab 16 Uhr vorgesehen. Falls es die Corona-Situation und die Teilnehmerzahl erlauben, gibt es am Anfang des neuen Jahres noch zwei Präsenztermine, in denen Stoff praktisch erarbeitet wird.

Modulteil: Softwaretechnik II (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Zugeordnete Lehrveranstaltungen:

Übung zu Softwaretechnik 2 (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Softwaretechnik II Klausur

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0130: Formale Methoden im Software Engineering <i>Formal Methods in Software Engineering</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
Lernziele/Kompetenzen: Fachbezogene Kompetenzen: <p>Die Studierenden können mathematisch-formale Methoden für die Programmverifikation, speziell bei sicherheitskritischer Software einsetzen. Durch das Verständnis von Kalkülen trainieren sie die Fertigkeit zum logischen, konzeptuellen und analytischen Denken. Sie können Spezifikationen von Datenstrukturen entwerfen und deren Eigenschaften formal beweisen. Sie sind in der Lage, funktionale Eigenschaften von sequentiellen und nebenläufigen Programmen zu formulieren und dafür Beweise zu erstellen. Sie erwerben die Fertigkeit zur abstrakten, modularen Modellierung von Softwaresystemen und zur systematischen Entwicklung von korrekten Programmen aus diesen Modellen. Sie erwerben dadurch die Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen und können mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen abgeben. Sie erwerben Fähigkeiten, um Beiträge zur Wissenschaft zu leisten.</p> Schlüsselqualifikationen: <ul style="list-style-type: none"> • Training des logischen Denkens • Analytisch-methodische Kompetenz • Erwerb von Abstraktionsfähigkeiten • Fähigkeiten zur Modellbildung für Softwaresysteme 		
Arbeitsaufwand: Gesamt: 240 Std. 60 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Formale Methoden im Software Engineering (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
Inhalte: <p>Übergeordnetes Ziel ist die Produktion beweisbar korrekter Software. In der Vorlesung werden verschiedene klassische Methoden für die Programmverifikation im Kleinen behandelt. Darüber hinaus werde innovative Techniken für die formale Modellierung und Verifikation großer Systeme vermittelt. Als Werkzeug kommt das KIV-System zum Einsatz, das die formale Spezifikation und Verifikation von Systemen ermöglicht. Konkrete Inhalte sind: Algebraische Spezifikationen, interaktives Theorembeweisen, Hoare-Logik, Dynamische Logik, Temporallogik</p>		

Literatur:

- Sperschneider, Antoniou: Logic: A Foundation for Computer Science, Addison Wesley 1991
- Loeckx, Ehrich, Wolf: Specification of Abstract Data Types, Wiley 1996
- Ausführliche Dokumentation
- Folienhandout

Modulteil: Formale Methoden im Software Engineering (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Prüfung

Formale Methoden im Software Engineering (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 45 Minuten

Modul INF-0131: Software- und Systemsicherheit <i>Software and Systems Security</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen:</p> <p>Fachbezogene Kompetenzen:</p> <p>Nach der erfolgreichen Teilnahme an der Lehrveranstaltung kennen Studierende die Grundlagen, Aufbau, und typische Schwächen kryptographischer Protokolle und praxisrelevanter, sicherheitskritischer Systeme, können diese verständlich und sicher darstellen, und verstehen die Grundprinzipien deren Sicherheitsanalyse.</p> <p>Sie können Bedrohungsanalyse durchführen, kryptographische Protokolle entwickeln, Chipkarten programmieren, sicherheitskritische Systeme in Teams entwerfen, und ihre Ergebnisse dokumentieren.</p> <p>Sie können systematisch Bedrohungen für Softwaresysteme analysieren und deren Risiken bewerten.</p> <p>Sie können einen modellgetriebenen Entwicklungsprozess für sicherheitskritische Systeme anwenden.</p> <p>Sie sind in der Lage, bedeutende technische Entwicklungen zu erkennen und fachliche Lösungskonzepte in Programme umzusetzen. Sie haben Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Abstraktionsfähigkeiten • Grundsätze guter wissenschaftlicher Praxis • Mathematisch-formale Grundlagen • Quantitative Aspekte der Informatik • Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen • Fähigkeit, vorhandenes Wissen selbstständig zu erweitern • Qualitätsbewusstsein und Akribie 		
<p>Arbeitsaufwand:</p> <p>Gesamt: 240 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>60 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig (i. d. R. im WS)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Moduleil: Software- und Systemsicherheit (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Deutsch</p> <p>SWS: 2</p>		

Inhalte:

Inhalt der Vorlesung ist der Entwurf sicherer Softwaresysteme, speziell verteilter Systeme, deren Sicherheit wesentlich auf dem Einsatz von Sicherheitsprotokollen beruht. Ein Schwerpunkt liegt dabei auf Anwendungen, in denen Chipkarten eingesetzt werden. In der Vorlesung werden Kenntnisse in JavaCard, der Chipkartentechnologie, Bedrohungsanalyse und dem Design kryptographischer Anwendungsprotokolle vermittelt, die in den Übungen an praktischen Beispielen (u.a. einer elektronischen Kopierkarte und einer elektronischen Fahrkarte) erprobt werden. Bei der Entwicklung der Protokolle wird der SecureMDD-Ansatz verwendet, eine Methode zur modellgetriebenen Entwicklung sicherheitskritischer Protokolle.

Literatur:

- Schneier: Applied Cryptography, Wiley and Sons, 1996 (2nd edition)
- Anderson, Needham: Programming Satan's Computer, in: Computer Science Today, Springer LNCS 1000, 1995
- Lowe: Breaking and fixing the Needham-Schroeder public-key protocol using FDR, in: Tools and Algorithms for the Construction and Analysis of Systems, Springer LNCS 1055, 1996
- Folienhandout, Spezifikationen und APIs

Modulteil: Software- und Systemsicherheit (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Prüfung

Software- und Systemsicherheit (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 45 Minuten

Modul INF-0133: Selbstorganisierende, adaptive Systeme <i>Self-organizing, adaptive systems</i>		8 ECTS/LP
Version 3.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Die Studierenden haben vertiefte Fachkenntnisse über die Eigenschaften und den Aufbau selbst-organisierender Systeme aus der Biologie, Soziologie, Physik und anderen Bereichen und der systematischen Modellierung und Konstruktion adaptiver Systeme in der Informatik und können solche Systeme analysieren und selbst entwerfen. Sie kennen Vor- und Nachteile verschiedener Entwurfsalternativen und können sie im Kontext der Problemstellung bewerten. Sie haben die Fertigkeit zum analytischen und konzeptionellen Denken und können geeignete Methoden auswählen und anwenden und wissenschaftlich aussagekräftige Bewertungen abgeben. Die Studierenden sind in der Lage, adaptive Systeme adäquat zu modellieren und dokumentieren.</p> <p>Außerdem kennen die Studierenden praxisrelevante Fragestellungen.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Erwerb von Abstraktionsfähigkeiten • Grundsätze guter wissenschaftlicher Praxis 		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vorlesung (Präsenzstudium) 60 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Selbstorganisierende, adaptive Systeme (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: In der Vorlesung werden die Grundlagen verschiedener Selbst-Organisationsmechanismen sowie das Handwerkszeug, um diese in IT-Systemen einsetzen zu können, vermittelt. Im Verlauf der Veranstaltung werden verschiedene Beispiele für selbstorganisierende Systeme vorgestellt, untersucht und Anwendungen der erlernten Organisationsprinzipien auf Beispiele aus der Informatik erläutert. Schließlich werden Methoden betrachtet, mit deren Hilfe sich Selbst-Organisation und Adaptivität in die Entwicklung komplexer Computersysteme integrieren lassen. Konkrete Themen sind: Selbst-Organisation, Emergenz, Chaostheorie, zelluläre Automaten, Spieltheorie, Multi-Agentensysteme, Autonomic Computing, Organic Computing.</p>		

Literatur:

- Gleick: Chaos: Making a New Science, Penguin 2008
- Strogatz: Sync : The Emerging Science of Spontaneous Order, Hyperion 2003
- Miller, Page: Complex Adaptive Systems: An Introduction to Computational Models of Social Life, Princeton University Press 2007
- Dawkins: The Selfish Gene, Oxford University Press, 3rd Revised Edition
- Wolfram: A New Kind of Science, Wolfram Media Inc. 2002
- von Neumann, Morgenstern: Theory of Games and Economic Behavior, 2004
- Folienhandout

Zugeordnete Lehrveranstaltungen:

Selbst-organisierende, adaptive Systeme (Vorlesung)

Veranstaltung wird online/digital abgehalten.

"Selbst-organisierende, adaptive Systeme" beschäftigt sich mit theoretischen Grundlagen, die für die Entwicklung von offenen Multiagentensystemen nötig sind. Dabei folgt die Vorlesung vor allem dem Aufbau von <http://www.masfoundations.org/> und behandelt Spieltheorie, Mechanism Design und (verteilte) Constraint-Optimierung. Sie richtet sich vor allem an den Anwendungsfällen des Lehrstuhls "Verteiltes Energiemanagement" sowie flexible Produktion (<https://www.uni-augsburg.de/de/fakultaet/fai/isse/projects/soproduction/>) und Planung und Selbstorganisation in mobilen Multi-Roboter-Systemen (<https://www.uni-augsburg.de/de/fakultaet/fai/isse/projects/combo/>) aus.

Modulteil: Selbstorganisierende, adaptive Systeme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Zugeordnete Lehrveranstaltungen:

Übung zu Selbst-organisierende, adaptive Systeme (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Selbstorganisierende, adaptive Systeme

Mündliche Prüfung / Prüfungsdauer: 45 Minuten

Beschreibung:

Ausnahmefall WiSe 2020/2021: Die Prüfung findet als Klausur statt.

Modul INF-0179: Einführung in die Spieleprogrammierung <i>Foundations of Game Programming</i>		8 ECTS/LP
Version 1.1.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Studierenden die wesentlichen Konzepte zur Entwicklung eines Spiels. Sie sind in der Lage, eigene Gameplay-Konzepte umzusetzen und passende KI-Verfahren zu Entscheidungsfindung sowie Ablaufsteuerung einzusetzen. Sie verfügen über die Kenntnisse der Denkweise und die Sprache anwendungsrelevanter Disziplinen. Im Rahmen der Vorlesung erlernen sie, KI-Komponenten mit geeigneten Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmik selbstständig zu entwickeln und technisch umzusetzen. Besonders gefördert wird in diesem Rahmen auch die Fertigkeit zur sicheren und überzeugenden Darstellung von Ideen und Konzepten, verständlichen Darstellung und Dokumentation von Ergebnissen sowie zum logischen, analytischen und konzeptionellen Denken.</p> <p>Schlüsselqualifikationen: Auswahl und sichere Anwendung geeigneter Methoden, Fachübergreifende Kenntnisse, Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen, Fertigkeit zur Entwicklung und Umsetzung von Lösungsstrategien für komplexe Probleme, Verstehen von Teamprozessen, Fertigkeit der Zusammenarbeit in Teams, Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen, Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen: Ferienaufgabe		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Einführung in die Spieleprogrammierung (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
Inhalte: Game Engines, Entscheidungsfindung für KI-Charaktere, Wegfindung und Navigation, Gruppenverhalten und Gruppendynamik, Animationen, Physik, Storytelling, Ein-/Ausgabemethodik, Audio, Game Design, Grafik und Shaderprogrammierung		
Literatur: Skript		
Modulteil: Einführung in die Spieleprogrammierung (Übung)		
Lehrformen: Übung		
Sprache: Deutsch		
SWS: 4		

Prüfung

Einführung in die Spieleprogrammierung

Portfolioprüfung

Beschreibung:

Ausnahmefall SoSe 2020: Portfolioprüfung

Modul INF-0182: Praktikum Multimodale Echtzeitsignalverarbeitung <i>Practical Module Multimodal Real Time Signal Processing</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierenden mit den Grundkonzepten der Signalverarbeitung und des maschinellen Lernens vertraut. Sie sind in der Lage, fachliche Lösungskonzepte in Programme und Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Im Rahmen des Praktikums erlernen sie außerdem, in kleinen Teams größere Projektaufgaben (Entwicklung von Softwaremodulen) zu planen, nach einem selbst entwickelten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und als Team zu präsentieren. Besonders gefördert wird zudem die Fähigkeit geeignete Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmik selbstständig zu entwickeln und technisch umzusetzen.</p> <p>Schlüsselqualifikationen: Fortgeschrittene Kenntnisse der Signalverarbeitung und des maschinellen Lernens, Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen, Fertigkeit zur Entwicklung und Umsetzung von Lösungsstrategien für komplexe Probleme, Verstehen von Teamprozessen, Fertigkeit der Zusammenarbeit in Teams, Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen, Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Praktikum (Präsenzstudium) 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
<p>Voraussetzungen: Programmiererfahrung Modul Multimedia Grundlagen I (INF-0087) - empfohlen Modul Multimedia Grundlagen II (INF-0166) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Praktikum Multimodale Echtzeitsignalverarbeitung Lehrformen: Praktikum Sprache: Deutsch SWS: 6</p>		
<p>Inhalte: Die konkrete Aufgabenstellung aus dem weiten Gebiet der multimodalen Echtzeitsignalverarbeitung wird jedes Jahr neu entworfen.</p>		
<p>Literatur: Literaturhinweise werden je nach Thema zu Beginn des Semesters bekanntgegeben.</p>		
<p>Prüfung Praktikum Multimodale Echtzeitsignalverarbeitung praktische Prüfung</p>		

Modul INF-0183: Praktikum Spieleprogrammierung <i>Practical Module Game Programming</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Studierenden die wesentlichen Konzepte der Spieleentwicklung. Sie sind in der Lage, in kleinen Teams größere Projektaufgaben zu planen, nach einem selbst entwickelten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren. Sie sind in der Lage, fachliche Lösungskonzepte in Programme und Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Sie verfügen über die Kenntnisse der Denkweise und die Sprache anwendungsrelevanter Disziplinen. Im Rahmen des Praktikums erlernen sie, verschiedenste Komponenten eines Spiels mit geeigneten Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmik selbstständig zu entwickeln und technisch umzusetzen. Besonders gefördert wird in diesem Rahmen auch die Fertigkeit zur sicheren und überzeugenden Darstellung von Ideen und Konzepten, verständlichen Darstellung und Dokumentation von Ergebnissen sowie zum logischen, analytischen und konzeptionellen Denken.</p> <p>Schlüsselqualifikationen: Fertigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten; Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen; Verstehen von Teamprozessen; Fertigkeit der Zusammenarbeit in Teams; Fähigkeit zur Leitung von Teams; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Fähigkeit, Beiträge zur Wissenschaft zu leisten; Kompetenz zum Erkennen von bedeutendetechnischen Entwicklungen; Qualitätsbewusstsein, Akribie.</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Praktikum (Präsenzstudium) 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: Modul Einführung in die Spieleprogrammierung (INF-0179) - empfohlen		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 2.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Praktikum Spieleprogrammierung Lehrformen: Praktikum Sprache: Deutsch SWS: 6</p>		
Inhalte: Innerhalb des Praktikums soll ein Spiel entwickelt werden. Der inhaltliche Schwerpunkt des Praktikums wird jedes Jahr neu festgelegt.		
Literatur: Literaturhinweise werden zu Beginn des Semesters bekanntgegeben.		
Zugeordnete Lehrveranstaltungen: Praktikum Spieleprogrammierung (Praktikum) <i>*Veranstaltung wird in Präsenz abgehalten.*</i> Informationen zur Veranstaltung finden Sie auf unserer Lehrstuhlwebsite: https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/hcm/studies/lehrveranstaltungen/praktikum-spieleprogrammierung/		

Prüfung

Praktikum Spieleprogrammierung

praktische Prüfung

Modul INF-0189: Qualitätssicherung im Software Engineering <i>Quality Assurance in Software Engineering</i>		5 ECTS/LP
Version 2.0.0 (seit SoSe18) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Nach erfolgreicher Absolvierung der Lehrveranstaltung verstehen Studierende die Grundtechniken der Qualitätssicherung und der Problemstellungen des Testens in der Praxis. Sie sind in der Lage, SW-Module zu spezifizieren und kennen die wesentlichen Testtechniken und deren Einsatzzwecke im Software Engineering. Sie können Testkonzepte für große Softwaresysteme entwerfen, modellieren und anwenden. Die Studierenden sind für das Thema Qualität im Software Engineering sensibilisiert und können verschiedene Qualitätskriterien/-metriken kritisch hinterfragen und bewerten.</p> <p>Des Weiteren kennen und verstehen sie die Prinzipien von konstruktiven Qualitätssicherungstechniken und -praktiken.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Training des logischen Denkens • Analytisch-methodische Kompetenz • Abwägen von Vorgehensweisen • Erwerb von Abstraktionsfähigkeiten • Verbesserung der eigenen Softwareentwicklungskompetenz 		
<p>Arbeitsaufwand: Gesamt: 150 Std. 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Qualitätssicherung im Software Engineering (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		

Inhalte:

Ingenieursdisziplinen kombinieren Design- und Entwicklungsaktivitäten mit Aktivitäten, die vorläufige und endgültige Produkte prüfen, um Mängel zu erkennen und zu beseitigen. Software Engineering ist hierbei keine Ausnahme: Konstruktion hochqualitativer Software bedarf einer sich ergänzenden Kombination von Maßnahmen des Designs und der Prüfung der Software über den gesamten Entwicklungszyklus hinweg. Gerade aufgrund der Durchdringung der Software von immer mehr kritischen Bereichen wie etwa Automotive oder Avionik rücken Maßnahmen zu Qualitätssicherung immer mehr in den Fokus der Aufmerksamkeit im modernen Software Engineering. In dieser Vorlesung werden Konzepte, Techniken und Methoden der Qualitätssicherung im Software Engineering vermittelt. Dies umfasst, u.a., die Spezifikation von Software in einem Kontinuum von natürlichsprachlicher bis formalsprachlicher Notation, automatisierte Methoden und Techniken zur analytischen sowie auch zur konstruktivistischen Qualitätssicherung, Entwicklung von Qualitätssicherungsstrategien sowie Grundlagen im Umgang mit gängigen Werkzeugen, die im Software Engineering zum Einsatz kommen. Den Abschluss bildet die kritische Auseinandersetzung mit formalen Methoden, die für besonders kritische Module zum Einsatz kommen können und in Zertifizierungsstandards anerkannt werden.

Literatur:

- P. Ammann und J. Offutt: Introduction to Software Testing. Cambridge University Press, 2008.
- M. Pezzè und M. Young: Software Testing and Analysis: Process, Principles, and Techniques. Wiley & Sons, 2008.
- R. Binder: Testing Object-Oriented Systems: Models, Patterns, and Tools. Addison-Wesley, 2000.
- M. Chemuturi: Mastering Software Quality Assurance: Best Practices, Tools and Techniques for Software Developers. J. Ross Publishing, 2011.
- G. O'Regan: Introduction to Software Quality. Springer, 2014.
- W. Reif: Software-Verifikation und ihre Anwendungen, it+ti Themenheft, Oldenbourg Verlag, 1997
- Vorlesungsskript
- In der Vorlesung bereitgestellte wiss. Publikationen, Journalartikel und Buchbeiträge.

Modulteil: Qualitätssicherung im Software Engineering (Übung)**Lehrformen:** Übung**Sprache:** Deutsch**SWS:** 2**Prüfung****Qualitätssicherung im Software Engineering (Mündliche Prüfung)**

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0207: Reinforcement Learning <i>Reinforcement Learning</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Studierenden die wesentlichen Konzepte des Reinforcement Learnings. Sie sind in der Lage, fachliche Lösungskonzepte in Programme und Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Sie verfügen über die Kenntnisse der Denkweise und die Sprache anwendungsrelevanter Disziplinen. Im Rahmen der Vorlesung erlernen sie, lernende Komponenten mit geeigneten Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmen selbstständig zu entwickeln und technisch umzusetzen. Besonders gefördert wird in diesem Rahmen auch die Fertigkeit zur sicheren und überzeugenden Darstellung von Ideen und Konzepten, verständlichen Darstellung und Dokumentation von Ergebnissen sowie zum logischen, analytischen und konzeptionellen Denken.</p> <p>Schlüsselqualifikationen: Fortgeschrittene mathematisch-formale Methodik, Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen, Fertigkeit zur Entwicklung und Umsetzung von Lösungsstrategien für komplexe Probleme, Verstehen von Teamprozessen, Fertigkeit der Zusammenarbeit in Teams, Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen, Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 60 Std. Übung (Präsenzstudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Reinforcement Learning (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
Inhalte: Markov-Entscheidungsproblem, Dynamische Programmierung, Monte Carlo Methoden, TD-Lernen, Eligibility Traces, Hierarchisches Bestärkendes Lernen, Planen und Lernen, Generalisierung		
Literatur: <ul style="list-style-type: none"> • Skript • Richard S. Sutton und Andrew G. Barto. Reinforcement Learning: An Introduction. MIT Press, 1998 		
Zugeordnete Lehrveranstaltungen:		
Reinforcement Learning (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i>		

Inhalt der Veranstaltung sind Methoden und Prinzipien des Reinforcement Learning. In der Übung werden verschiedene Probleme modelliert, mit Hilfe von Lernalgorithmen als Simulationen implementiert und verglichen. Siehe <https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/hcm/studies/lehveranstaltungen/reinforcement-learning/>

Modulteil: Reinforcement Learning (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Zugeordnete Lehrveranstaltungen:

Übung zu Reinforcement Learning (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Reinforcement Learning

Portfolioprüfung

Modul INF-0213: Parallele und Verteilte Datenbanksysteme <i>Parallel and Distributed Database Systems</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe16) Modulverantwortliche/r: PD Dr. Markus Endres		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an der Veranstaltung sind die Studierenden in der Lage die Konzepte und Methoden, Verfahren, Techniken und Technologien von parallelen und verteilten Datenbanksystemen im Detail zu verstehen, zu bewerten und anzuwenden. Darüber hinaus können die Studierenden weiterführende komplexe Problemstellungen auf dem Gebiet Datenbanken, insbesondere unter Verwendung von parallelen und verteilten Algorithmen, analysieren und lösen. Die Vorlesung vermittelt zudem die Fertigkeit zum logischen, analytischen und konzeptionellen Denken. Bei den in der Veranstaltung anfallenden praxisnahen Programmieraufgaben ist eine sichere und überzeugende Darstellung von Ideen und Konzepten sowie die Zusammenarbeit in Teams entscheidend.</p> <p>Schlüsselqualifikationen: Fachspezifische Vertiefung; Kenntnisse der Vor-/Nachteile von Entwurfalternativen und deren Bewertung im jeweiligen Anwendungszusammenhang; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen; Fähigkeit, vorhandenes Wissen selbständig zu erweitern;</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 3.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Parallele und Verteilte Datenbanksysteme (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Vorlesung behandelt parallele und verteilte Datenbanksysteme. Dazu wird die Architektur von parallelen und verteilten Datenbanksystemen vorgestellt. Der Schwerpunkt liegt anschließend bei der Ausarbeitung und Implementierung von parallelen und verteilten Algorithmen. Insbesondere werden hierbei die parallele Suche, parallele Gruppierung, parallele Joins, Transaktionen in verteilten Datenbanken, verteilte Transaktionsprotokolle, paralleles OLAP, paralleles Data Mining und paralleles Clustering und Klassifikation behandelt.</p>		

Literatur:

- - Taniar et al.: High-Performance Parallel Database Processing and Grid Databases
- - E. Rahm: Mehrrechner-Datenbanksysteme
- - M. T. Özsu, P. Valduriez: Principles of Distributed Database Systems
- - P. Dada: Verteilte Datenbanken und Client/Server-Systeme
- - S. Conrad: Förderierte Datenbanksysteme
- - S. K. Rahimi, F. S. Haug: Distributed Database Management Systems - A Practical Approach

Modulteil: Parallele und Verteilte Datenbanksysteme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Parallele und Verteilte Datenbanksysteme (mdl. Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0217: Praktikum Autonomes Fahren <i>Practical Module Autonomous Driving</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe16) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Praktikum Autonomes Fahren verstehen es die Studierenden praxisnahe Problemstellungen hoher Komplexität im Bereich der Konzeptionierung, Entwicklung und Absicherung von hochautomatisierten/ autonomen Fahrzeugen mit aktuellen Methoden und Tools der modellbasierten Entwicklung zu lösen. Die Studierenden erlangen tiefgehende fachspezifische als auch fächerübergreifende Kenntnisse und Fähigkeiten, beispielsweise aus der Hardwarenahen Informatik, dem Software Engineering, als auch der zugrundeliegenden Fahrphysik und Mathematik. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren, zu präsentieren und verständlich zu dokumentieren.</p> <p>Schlüsselqualifikation: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Projektmanagementfähigkeiten</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 150 Std. Praktikum (Präsenzstudium)</p>		
<p>Voraussetzungen: Empfohlen wird die Teilnahme an einem der beiden Seminare. Modul Seminar Grundlagen des Software Engineering für Automotive Systems (BA) (INF-0027) - empfohlen Modul Seminar Grundlagen des Software Engineering für Automotive Systems (MA) (INF-0040) - empfohlen</p>		
<p>Angebotshäufigkeit: jedes Sommersemester</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 10</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p> <p>Modulteil: Praktikum Autonomes Fahren Lehrformen: Praktikum Sprache: Deutsch / Englisch SWS: 10</p>		

Inhalte:

In diesem Praktikum lernen die Teilnehmer, wie verschiedene ausgewählte Teilaspekte des Autonomen Fahrens umgesetzt, simuliert und analysiert werden können.

In einem Einführungskurs werden die benötigten theoretischen Grundlagen für die Bearbeitung der Praxisaufgaben gelegt.

Darüber hinaus lernen die Teilnehmer u.a. das im Automotive-Umfeld häufig eingesetzte Entwicklungswerkzeug „ADTF“ von Elektrobit und die darin verwendete Programmiersprache C++ kennen.

Nach dem Einführungskurs sollen in Gruppen von zwei bis drei Teilnehmern mithilfe der genannten Werkzeuge autonome Fahrfunktionen umgesetzt werden.

Die entwickelten Ergebnisse werden abschließend an 1:8 Fahrzeugmodellen, deren Sensorik realen Fahrzeugen sehr nahe kommt, demonstriert und ausgewertet.

Prüfung

Praktikum Autonomes Fahren

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0233: Industrierobotik <i>Industrial Robotics</i>		8 ECTS/LP
Version 2.2.0 (seit SoSe17) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Die Studierenden sind in der Lage Industrieroboter zu programmieren, kennen das dafür erforderliche methodische Vorgehen und können die Eignung verschiedener Vorgehensweise bewerten. Sie können fachliche Lösungskonzepte in Roboterprogramme umsetzen und dabei Entwurfsalternativen bewerten und anwenden. Sie haben Kenntnisse von praxisrelevanten Aufgabenstellungen in der automatisierten Fertigung. Sie haben die Fertigkeit zum analytischen und konzeptionellen Denken.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Erwerb von Abstraktionsfähigkeiten 		
<p>Bemerkung: Dieses Modul ersetzt das Modul "INF-0132: Software in Mechatronik und Robotik". Sofern dieses bereits belegt wurde, ist eine erneute Belegung nicht möglich.</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile
<p>Modulteil: Industrierobotik (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>
<p>Inhalte: Ziel der Veranstaltung ist es, an Beispielen die Programmierung und der Entwurf von Software für Industrieroboter, wie sie z.B. in der Automobilindustrie verwendet werden, zu erlernen. Dazu werden im ersten Teil der Vorlesung Grundlagen wie Kinematik und Bahnplanung mit Hilfe simulationsbasierter Programmieransätze behandelt. Im zweiten Teil der Vorlesung werden verschiedene, kleine Programmieraufgabenstellungen bearbeitet und auf einem KUKA Roboter evaluiert. Die Programmierung erfolgt mit der Roboterprogrammiersprache KRL.</p>
<p>Literatur:</p> <ul style="list-style-type: none"> • L. Sciavicco, B. Siciliano: Modelling and Control of Robot Manipulators. Reihe: Advanced Textbooks in Control and Signal Processing. Springer 2000 (2nd Ed.) • Handbücher von KUKA • Folienhandout

Zugeordnete Lehrveranstaltungen:

Industrierobotik (Vorlesung)

Veranstaltung wird als Hybrid/gemischt abgehalten.

Diese Veranstaltung steht unter dem Motto "Uni goes Industry", da in dieser Veranstaltung die Programmierung "echter" Industrieroboter incl. zugehöriger Software-Entwicklungsumgebung und Simulationsmöglichkeiten vermittelt wird. Dazu werden in Zweiergruppen verschiedene, kleine Programmieraufgabenstellungen bearbeitet und auf einem KUKA KR 6 Roboter evaluiert. Zudem werden wichtige Grundlagen der Robotik wie Kinematik und Bahnplanung anhand eines simulierten Roboters behandelt. Ziel der Veranstaltung ist es, an Beispielen die Programmierung und der Entwurf von Software für Industrieroboter, wie sie z.B. in der Automobilindustrie verwendet werden, zu erlernen. Die Vorlesung "Industrierobotik" ersetzt die Veranstaltung "Software in Mechatronik und Robotik". Studierende, die an der Vorlesung "Software in Mechatronik und Robotik" teilgenommen haben, können diese Veranstaltung nicht mehr besuchen.

... (weiter siehe Digicampus)

Modulteil: Industrierobotik (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Zugeordnete Lehrveranstaltungen:

Übung zu Industrierobotik (Übung)

Veranstaltung wird als Hybrid/gemischt abgehalten.

Prüfung

Industrierobotik

praktische Prüfung / Prüfungsdauer: 45 Minuten

Bearbeitungsfrist: 3 Monate

Beschreibung:

Ausnahmefall WiSe 2020/2021: Die Prüfung findet als Portfolioprüfung statt.

Modul INF-0235: Software für Industrie 4.0 <i>Software for Industry 4.0</i>		5 ECTS/LP
Version 1.2.0 (seit WS16/17) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Die Studierenden sind in der Lage die praktischen und methodischen Herausforderungen, die Industrie 4.0 für die Automatisierung und deren Softwareentwicklung bedeutet, zu verstehen. Sie werden sowohl in Techniken der klassischen Automatisierung als auch modernen Softwaretechnologien im Umfeld von Industrie 4.0 eingeführt und kennen so unterschiedliche anwendungsrelevante Disziplinen. Sie sind in der Lage Lösungskonzepte zu erstellen und dabei Entwurfsalternativen bewerten und anwenden.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenzen • Abwägen von Lösungsansätzen • Erwerb von Abstraktionsfähigkeiten 		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Software für Industrie 4.0 (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		
<p>Inhalte: Die Vorlesung behandelt aktuelle Themen im Kontext von Industrie 4.0. Dazu zählen folgende Inhalte:</p> <ul style="list-style-type: none"> • Einführung in die Automatisierung • Referenzarchitekturen für Industrie 4.0 • Einführung in die (mobile Service) Robotik • OPC UA • AutomationML • Data Analytics für Industrie 4.0 		
<p>Literatur:</p> <ul style="list-style-type: none"> • Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0 - Abschlussbericht des Arbeitskreises Industrie 4.0; acaTech 		

Modulteil: Software für Industrie 4.0 (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Software für Industrie 4.0

Mündliche Prüfung / Prüfungsdauer: 45 Minuten

Modul INF-0248: Kollaborative Robotik <i>Collaborative Robotics</i>		8 ECTS/LP
Version 1.1.0 (seit SoSe17) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Die Studierenden sind in der Lage kollaborative Roboter zu programmieren und verstehen die regulatorischen Randbedingungen (vgl. ISO/TS 15066:2016). Sie verstehen die grundlegenden Konzepte und Methoden der kollaborativen Robotik. Sie können dementsprechend fachliche Lösungskonzepte umsetzen und Entwurfsalternativen bewerten und anwenden. Sie haben die Fertigkeit zum analytischen und konzeptionellen Denken und können grundlegende Bewertungen der Risiken eines kollaborativen Robotersystems durchführen. Die Studierenden können abwägen, für welche praxisrelevanten Problemstellungen kollaborative Roboter eingesetzt werden können.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Erwerb von Abstraktionsfähigkeiten • Erwerb von Grundlagen für die Sicherheitsbewertung 		
<p>Bemerkung: Voraussetzungen (empfohlen): Software für Industrie 4.0 oder Industrierobotik oder Grundlagen der Robotik</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std.</p>		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Kollaborative Robotik (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 4</p>		

Inhalte:

Die Digitalisierung dringt mittlerweile in viele Bereiche des täglichen Lebens vor. Von Industriemaschinen und Robotersystemen (*Smart Factory, Industrie 4.0*) bis hin zu intelligenten Hausgeräten oder Heizungsanlagen (*Smart Home*) findet eine immer stärkere Vernetzung dieser Geräte im Internet der Dinge statt. Hierbei spricht man von cyber-physischen Systemen, die einerseits durch ein komplexes Zusammenspiel von vernetzten eingebetteten Systemen entstehen und andererseits geprägt sein werden von einer vollkommen neuen Art der Mensch-Technik-Interaktion in den Anwendungen.

Eine besondere Rolle unter den cyber-physischen Systemen nehmen Assistenz- oder Serviceroboter ein, da sie durch einen geteilten Arbeitsraum in eine direkte, auch physische Interaktion mit dem Menschen treten können. Diese robotischen Systeme können den Menschen in seiner täglichen Arbeit unterstützen bzw. entlasten. Das gilt sowohl im häuslichen Umfeld als auch in der Fabrik. Im Kontext von Industrie 4.0 haben robotische Co-Worker (CoBots) vereinzelt bereits Eingang in die Produktion gefunden. Diese kollaborativen Einsatzszenarien werden jedoch in Zukunft noch weiter an Bedeutung gewinnen. Durch die in 2016 veröffentlichte Norm ISO/TS 15066:2016 zur Mensch-Roboter-Kollaboration stehen dem Einsatz innovativer Robotersysteme neue Möglichkeiten offen.

Während die grundlegende Hardware, d.h. die Roboter, mobilen Plattformen, Greifer und Sensoren, bereitgestellt wird, liegt der Schwerpunkt der Veranstaltung auf der Entwicklung von Software für kollaborative (mobile) Manipulatoren. Durch die intelligente Vernetzung von Aktuatorik und Sensorik, neuartige Algorithmen für die Sensorauswertung und kreative Konzepte zur (physischen) Interaktion mit dem Menschen entstehen innovative, kollaborative Robotersysteme im praktischen Teil der Veranstaltung.

Infolgedessen liegt der thematische Fokus der Veranstaltung auf

- der Umsetzung multi-modaler Benutzerschnittstellen,
- der Gestaltung neuartiger, auch physischer Interaktionsmöglichkeiten,
- der Realisierung intelligenter Verhaltensweisen durch Techniken der Künstlichen Intelligenz bzw. des Machine Learnings und
- der Entwicklung kognitiver Fähigkeiten des Robotersystems.

Diese Kernelemente bieten die Grundlage, um eine praktische Aufgabenstellung prototypisch umzusetzen. Die Programmierung erfolgt dabei in objektorientierten Programmiersprachen. So können kollaborative Roboter bspw. in Java mit der Robotics API programmiert werden, einem am ISSE in Kooperation mit KUKA entwickelten Roboterframework. Alternativ oder ergänzend kann das Robot Operating System (ROS) verwendet werden, das in den Programmiersprachen C++ und Python programmiert werden kann.

Zugeordnete Lehrveranstaltungen:**Übung zu Kollaborative Robotik (Übung)**

Veranstaltung wird als Hybrid/gemischt abgehalten.

Modulteil: Kollaborative Robotik (Vorlesung)

Lehrformen: Vorlesung

Sprache: Deutsch

SWS: 2

Literatur:

- ISO/TS 15066:2016. 2016-02-15. Robots and robotic devices - Collaborative robots
- DIN EN ISO 13849-1:2016-06. 06.2016. Sicherheit von Maschinen - Sicherheitsbezogene Teile von Steuerungen - Teil 1: Allgemeine Gestaltungsleitsätze (ISO 13849-1:2015); Deutsche Fassung EN ISO 13849-1:2015
- DIN EN ISO 10218-1:2012. 01.2012. Industrieroboter - Sicherheitsanforderungen - Teil 1: Roboter (ISO 10218-1:2011); Deutsche Fassung EN ISO 10218-1:2011

Zugeordnete Lehrveranstaltungen:**Kollaborative Robotik (Vorlesung)**

Veranstaltung wird als Hybrid/gemischt abgehalten.

Der Schwerpunkt der Veranstaltung liegt auf der Entwicklung von Software für kollaborative (mobile) Manipulatoren; die grundlegende Hardware, d.h. die Roboter, mobilen Plattformen, Greifer und Sensoren, wird bereitgestellt. Durch die intelligente Vernetzung von Aktuatorik und Sensorik, neuartige Algorithmen für die Sensorauswertung und kreative Konzepte zur (physischen) Interaktion mit dem Menschen entstehen innovative, kollaborative Robotersysteme im praktischen Teil der Veranstaltung. Infolgedessen liegt der thematische Fokus der Veranstaltung auf • der Umsetzung multi-modaler Benutzerschnittstellen, • der Gestaltung neuartiger, auch physischer Interaktionsmöglichkeiten, • der Realisierung intelligenter Verhaltensweisen durch Techniken der Künstlichen Intelligenz bzw. des Machine Learnings und • der Entwicklung kognitiver Fähigkeiten des Robotersystems. Diese Kernelemente bieten die Grundlage, um eine praktische Aufgabenstellung prototypisch umzusetzen. Die Programmierung erfolgt dab
... (weiter siehe Digicampus)

Prüfung

Kollaborative Robotik

praktische Prüfung / Prüfungsdauer: 60 Minuten

Bearbeitungsfrist: 3 Monate

Modul INF-0250: Praktikum Reinforcement Learning <i>Practical Module Reinforcement Learning</i>		8 ECTS/LP
Version 1.3.0 (seit SoSe17) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Studierenden die wesentlichen Konzepte des Reinforcement Learnings. Sie sind in der Lage, in kleinen Teams größere Projektaufgaben zu planen, nach einem selbst entwickelten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren. Sie sind in der Lage, fachliche Lösungskonzepte in Programme und Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Sie verfügen über die Kenntnisse der Denkweise und die Sprache anwendungsrelevanter Disziplinen. Im Rahmen des Praktikums erlernen sie, lernende Komponenten mit geeigneten Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmen selbstständig zu entwickeln und technisch umzusetzen. Besonders gefördert wird in diesem Rahmen auch die Fertigkeit zur sicheren und überzeugenden Darstellung von Ideen und Konzepten, verständlichen Darstellung und Dokumentation von Ergebnissen sowie zum logischen, analytischen und konzeptionellen Denken.</p> <p>Schlüsselqualifikationen: Fertigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten; Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen; Verstehen von Teamprozessen; Fertigkeit der Zusammenarbeit in Teams; Fähigkeit zur Leitung von Teams; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Fähigkeit, Beiträge zur Wissenschaft zu leisten; Kompetenz zum Erkennen von bedeutendetechnischen Entwicklungen; Qualitätsbewusstsein, Akribie.</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 90 Std. Praktikum (Präsenzstudium)</p>		
<p>Voraussetzungen: Programmiererfahrung Modul Reinforcement Learning (INF-0207) - empfohlen</p>		
<p>Angebotshäufigkeit: jedes Sommersemester</p>	<p>Empfohlenes Fachsemester: ab dem 2.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 6</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p>		
<p>Modulteil: Praktikum Reinforcement Learning Lehrformen: Praktikum Sprache: Deutsch SWS: 6</p>		
<p>Inhalte: Die konkrete Aufgabenstellung aus dem Gebiet "Reinforcement Learning" wird jedes Semester neu entworfen.</p>		
<p>Prüfung Praktikum Reinforcement Learning praktische Prüfung Beschreibung: Ausnahmefall SoSe 2020: Praktische Prüfung</p>		

Modul INF-0272: Intelligente Signalanalyse in der Medizin <i>Intelligent Signal Analysis in Medicine</i>		5 ECTS/LP
Version 1.1.0 (seit WS17/18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Knowledge: The students learn the principal concepts of sequential signal processing, signal source separation, and feature extraction and information reduction exemplified by medically relevant audio and bio signals. They further gain insight into machine learning principles such as learning dynamics and context as is needed for many intelligent signal analysis tasks. They will learn about different problems and solutions in the analysis of a variety of signals relevant in the context of health care, wellbeing, and general medical signals analysis. Students will get to know the mindset of modern machine learning, computer-aided health care, and get to know ethical implications.</p> <p>Skills: The students will be able to choose appropriate algorithms of signal processing and machine intelligence, further develop these, design new solutions, and apply these to a broad range of medical signal analysis problems. They will practice to think logically and conceptionally in order to select appropriate solutions to a given task. Students will be able to recognise important technical developments in the field of signal processing, machine learning and e-Health/m-Health.</p> <p>Competences: The students are able to characterise, judge on the quality and suitability, and design suited algorithmic solutions for intelligent signal analysis with a focus on medical signals. They are further able to realise the learnt concepts in programs and machine learning models. Participants will be able to analyse and structure complex and practice-oriented problems in the field of m-Health and e-Health and to find suitable and state-of-the-art solutions. They know how to make scientifically meaningful evaluations of proposed systems. They will further learn how to document and present results in a reasonable and meaningful way.</p> <p>Key skills: Formal methods; Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality awareness; Scientific working; Literature research.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>30 Std. Übung (Präsenzstudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
<p>Voraussetzungen:</p> <p>Knowledge of basic mathematic lectures should be present.</p>		
<p>Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 4</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p> <p>Modulteil: Intelligente Signalanalyse in der Medizin (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Englisch</p> <p>SWS: 2</p>		

Inhalte:

Topics: Basics of Signal Processing, Signal Source Separation, Data Acquisition and Annotation, Audio-Visual Feature Extraction, Machine Learning, e-Health, m-Health, Ethics, Python, Machine Learning Toolkits.

Literatur:

Björn Schuller: „*Intelligent Audio Analysis*“, Signals and Communication Technology, Springer, ISBN: 978-3642368059, 2013.

Modulteil: Intelligente Signalanalyse in der Medizin (Übung)

Lehrformen: Übung

Sprache: Englisch

SWS: 2

Prüfung

Intelligente Signalanalyse in der Medizin (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0273: Praktikum Mobile Sensing for Fitness and Well-being <i>Practical Module Sensing for Fitness and Wellbeing</i>		5 ECTS/LP
Version 1.0.0 (seit WS17/18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Students learn to realise concepts and models of sensing health- and fitness-related parameters on smart devices. They learn how to acquire signals from different modalities and sensors and to implement algorithms of pattern recognition on mobile devices. After participation in the Praktikum, students know how to analyse and structure complex problems in the field and to select suitable and state-of-the-art approaches to their solution.</p> <p>Participants are trained in their analytical and conceptional skills as well as in practical programming skills to transfer their knowledge to a practical task. They learn how to make scientifically meaningful assessments of their system using appropriate methods. They will know the mindset of two different fields, software development and machine learning. All knowledge obtained during the Praktikum is applied in practice-oriented tasks. Furthermore, after participation, students will be able to recognise important technical evolution in the field of sensors, mobile apps and related tools, and intelligent signal analysis.</p> <p>Students will work in teams and organise their work and task distribution in an autonomous way. They will learn how to summarise, present and document results in a reasonable way</p> <p>Key skills: Formal methods; Methods for software development and abstraction; Versioning tools; Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams, Understanding of team management; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality Awareness.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std. 60 Std. Praktikum (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Mobile Sensing for Fitness and Wellbeing		
Lehrformen: Praktikum		
Sprache: Englisch		
Angebotshäufigkeit: jedes Wintersemester		
SWS: 4		
<p>Inhalte:</p> <p>Research in the field of m-Health is focussed on the design and development of sensors, systems, and applications to recognise, interpret and simulate human states w.r.t. fitness, health, and wellbeing. In this Praktikum, students will experience in designing relevant systems, which are using modalities originating from different sensors, such as, vital signs, audio, speech, and video. In small teams, they will implement and evaluate an application running on a smart device.</p>		

Literatur:

Wird vom Dozenten / von der Dozentin bekannt gegeben

Prüfung

Praktikum Mobile Sensing for Fitness and Wellbeing

Praktikum

Modul INF-0277: Analyzing Massive Data Sets <i>Analyzing Massive Data Sets</i>		8 ECTS/LP
Version 1.2.0 (seit SoSe18) Modulverantwortliche/r: Prof. Dr. Peter Michael Fischer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an der Veranstaltung sind die Studierenden in der Lage die Konzepte und Methoden, Verfahren, Techniken und Technologien zur Analyse von massiv großen Datensätzen zu verstehen und zu bewerten. Mögliche Inhalte umfassen:</p> <ul style="list-style-type: none"> • Grundlagen des Information Retrieval • Ähnlichkeitssuche und Clustering • Analyse von Datenströmen und temporalen Daten • Webgraphen: Linkanalyse und soziale Netzwerke • Dynamische Netzwerke und Informationsausbreitung • Empfehlungssysteme und Onlinewerbung • Berechnungsverfahren für massive Datensätze <p>Außerdem können die Studierenden fachliche Lösungskonzepte zur Analyse großer Datensätze in Programme umsetzen.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken, Abwägen von Lösungsansätzen, Erwerb von Abstraktionsfähigkeiten; Fachspezifische Vertiefung; Umsetzen fachlicher Lösungskonzepte in Programme und Modelle; Kenntnisse der Vor-/Nachteile von Entwurfsalternativen und Bewertung im jeweiligen Anwendungszusammenhang; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit, mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen anzustellen; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen;</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen: Modul Datenbanksysteme (INF-0073) - empfohlen Modul Diskrete Strukturen für Informatiker (INF-0109) - empfohlen Modul Informatik 3 (INF-0111) - empfohlen</p>		
<p>Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 6</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p> <p>Modulteil: Analyzing Massive Data Sets (Vorlesung) Lehrformen: Vorlesung Sprache: Englisch SWS: 4</p>		

Inhalte:

Die Vorlesung behandelt grundlegende Konzepte zur Analyse massiv großer Datensätze wie Informationsextraktion, Ähnlichkeitssuche, Clustering, Link- und Netzwerkanalyse sowie deren Implementierung.

Literatur:

- Mining of Massive Datasets. J. Leskovec, A. Rajaraman, J.D. Ullman. Cambridge University Press, 2014
- D. Easley, J. Kleinberg. Networks, Crowds, and Markets: Reasoning About a Highly Connected World. Cambridge University Press, 2010.
- R. Baeza-Yates, B. Ribeiro-Neto: Modern Information Retrieval

Weitere Literatur wird in der Vorlesung bekannt gegeben

Modulteil: Analyzing Massive Data Sets (Übung)

Lehrformen: Übung

Sprache: Englisch / Deutsch

SWS: 2

Prüfung

Analyzing Massive Data Sets

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0279: Music Informatics <i>Music Informatics</i>		5 ECTS/LP
Version 1.3.0 (seit SoSe18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Knowledge: The course Music Informatics presents the fundamental concepts of music theory and the music language and its representation in the visual, symbolic, and acoustic domain. Several digital formats for music symbolic representation, such as Music XML, MEI, Kern**, and MIDI protocol, as well as open source tools such as LilyPond and Csound will be introduced. Machine learning principles and techniques with applications in music information retrieval and computational musicology will be practically applied. Students will learn about different problems and solutions in the analysis of symbolic and acoustic music data. Students will get to know the mindset from both sides, the musicological and the computer scientist perspective.</p> <p>Skills: The students will understand the basic principles of music theory and its representation in digital language, being able to analyse, interpret, and create musical samples in a variety of symbolic formats and programming languages. They will learn to apply machine learning procedures, such as feature extraction and pattern recognition, to music information retrieval problems, such as key detection and music-score synchronisation, amongst other. After participation, students will know how to advance existing concepts and approaches in the field of music informatics and data analysis. Furthermore, they will be able to recognise important technical developments in the field of data science and signal processing.</p> <p>Competences: By integrating basic principles of music theory, its representation in digital language, and machine learning techniques, the students will be able to identify new problems and solutions in the field of music information retrieval considering a variety of musical styles and genres. The students are able to characterise, judge on the quality and suitability, and design suited algorithmic solutions for music data analysis in both the symbolic and the audio domain.</p> <p>Participants will be able to analyse and structure complex and practice-oriented problems in the field of music informatics and to find suitable solutions, by using state-of-the-art tools and complementary methods, if needed. They know how to make scientifically meaningful evaluations of proposed systems. They will further learn how to document and present results in a reasonable and meaningful way.</p> <p>Key skills: Computational musicology, Music theory, Digital Music Representation, Basics of Signal Processing, Machine Learning, Music Information Retrieval, Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality awareness; Scientific working; Literature research.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>30 Std. Übung (Präsenzstudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen:		
Knowledge of basic mathematic lectures should be present		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	

<p>Modulteile</p>
<p>Modulteil: Music Informatics (Lecture) Lehrformen: Vorlesung Sprache: Englisch Angebotshäufigkeit: jedes Sommersemester SWS: 2</p>
<p>Inhalte:</p> <p>In Music Informatics, the basic principles of music theory will be presented from both the traditional and computational point of view. Music will be evaluated in three domains: visual, symbolic, and acoustic; and for each of them: formats, programming languages, and machine learning tools will be studied. This course will give a basic introduction to music information retrieval and computational musicology by identify problems and solutions for different kinds of musical genres and styles.</p>
<p>Literatur:</p> <ul style="list-style-type: none"> • Meinard Müller: <i>"Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications."</i> Springer, ISBN: 978-3-319-21944-8. 2015. • Björn Schuller: <i>"Intelligent Audio Analysis"</i>, Signals and Communication Technology, Springer, ISBN: 978-3642368059, 2013.
<p>Modulteil: Music Informatics (Tutorial) Lehrformen: Übung Sprache: Englisch Angebotshäufigkeit: jedes Sommersemester SWS: 2</p>
<p>Prüfung Music Informatics (Exam) Klausur / Prüfungsdauer: 90 Minuten</p>

Modul INF-0284: Praktikum Mobile Application Development <i>Practical Module Mobile Application Development</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Students learn to realise concepts and models of mobile application development on the Android platform. They learn how to acquire data and signals from different sensors and to implement algorithms of pattern recognition and data/signal analysis on mobile devices. After participation in the Praktikum, students know how to analyse and structure complex problems in the field and to select suitable and state-of-the-art approaches to their solution.</p> <p>Participants are trained in their analytical and conceptional skills as well as in practical programming skills to transfer their knowledge to a practical task. They learn how to make scientifically meaningful assessments of their system using appropriate methods. They will know the mindset and thinking of application development and software engineering. All knowledge obtained during the Praktikum is applied in practice-oriented tasks. Furthermore, after participation, students will be able to recognise important technical evolution in the field of sensors, mobile apps and related tools, and smart devices.</p> <p>Students will work in teams and organise their work and task distribution in an autonomous way. They will learn how to summarise, present and document results in a reasonable way.</p> <p>Key skills: Formal methods; Methods for software development and abstraction; Versioning tools; Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams, Understanding of team management; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality Awareness.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std. 60 Std. Praktikum (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: Programming skills in Java are required.		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Mobile Application Development		
Lehrformen: Praktikum		
Sprache: Englisch		
SWS: 4		
<p>Inhalte:</p> <p>In the Praktikum Mobile Application Development, students will develop software for the Android platform in groups of 2-4 students.</p> <p>Students will get to know and to program several groups of Android devices, e.g., mobiles and wearables, and respective sensors. They will familiarise with the Android Studio IDE, the according toolchain and the code management tool Gi.</p> <p>Topics: Java, Android, App development, UI design, Audio Player, Map and Location, Camera control, Mobile devices, Intelligent signal processing, Wearables.</p>		

Literatur:

The literature is going to be announced by the lecturers during the course.

Prüfung

Praktikum Mobile Application Development

Praktikum

Modul INF-0291: Praktikum Selbstlernende Systeme <i>Practical Module Self-Learning Systems</i>		8 ECTS/LP
Version 1.2.0 (seit WS18/19) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
Lernziele/Kompetenzen: Nach der Teilnahme am Praktikum sind die Studierenden in der Lage, Problemstellungen (höherer Komplexität) in dem Bereich „maschinelle Lernverfahren“ zu verstehen und zu lösen. Sie können unterschiedliche Verfahren vergleichen und einordnen und diese eigenständig auf konkrete Beispiele aus der Praxis anwenden. Studenten können intelligente Systeme im Bezug auf die algorithmische Lösung bewerten und sind weiterhin mit Verfahren zur Leistungsevaluierung eines intelligenten Systems vertraut. Sie sind außerdem in der Lage, in kleinen Teams, größere Projektaufgaben (Entwicklung von Softwaremodulen) zu planen, nach einem selbst entwickelten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und als Team zu präsentieren. Weiterhin können sie bedeutende technische Enticklungen im Bereich „maschinelles Lernen“ erkennen und einordnen.		
Schlüsselqualifikationen: <ul style="list-style-type: none"> - Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen, Fertigkeiten zur Entwicklung und Umsetzung von Lösungsstrategien - Kompetenz zur Vernetzung unterschiedlicher Fachgebiete - Präsentation und Dokumentation (eigener) Ergebnisse - Analytische methodische Kompetenz - Fähigkeit produktiv und zielführend im Team zu arbeiten - Akribisches Arbeiten - Fachübergreifende Kenntnisse - Systematische Weiterentwicklung von Entwurfsmodellen - Zeitmanagement - Eigenständige Literaturrecherche zu angrenzenden Themen - Grundsätze guter wissenschaftlicher Praxis 		
Arbeitsaufwand: Gesamt: 240 Std. 225 Std. Praktikum (Selbststudium) 15 Std. Teilnahme an Lehrveranstaltungen (Präsenzstudium)		
Voraussetzungen: Empfohlen: die erfolgreiche Teilnahme an der Vorlesung OC2, Programmiererfahrung, Teamfähigkeit Modul Organic Computing II (INF-0066) - empfohlen		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 2.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Selbstlernende Systeme Lehrformen: Praktikum Dozenten: Prof. Dr. Jörg Hähner Sprache: Deutsch SWS: 2 ECTS/LP: 8.0		

Inhalte:

In dem Praktikum „Selbstlernende Systeme“ sollen die Studenten unterschiedliche Methoden aus dem Bereich „machine learning“ kennenlernen und vor allem selber implementieren. In einem Einführungskurs werden die benötigten theoretischen Grundlagen gelegt und diese dann innerhalb von Kleingruppen an konkreten Beispielen aus der Praxis angewandt.

Literatur:

aktuelle wissenschaftliche Paper

Zugeordnete Lehrveranstaltungen:

Praktikum zu Selbstlernende Systeme (Praktikum)

Veranstaltung wird online/digital abgehalten.

In dem Praktikum „Selbstlernende Systeme“ sollen die Studenten unterschiedliche Methoden aus dem Bereich „machine learning“ kennenlernen und vor allem selber implementieren. In einem Einführungskurs werden die benötigten theoretischen Grundlagen gelegt und diese dann innerhalb von Kleingruppen an konkreten Beispielen aus der Praxis angewandt.

Prüfung

Praktikum Selbstlernende Systeme

Portfolioprüfung, Kombination aus Praktischer und Schriftlich-Mündlicher Prüfung

Beschreibung:

schriftliche Abgaben, Softwareabnahme, Abschlussvortrag

Modul INF-0293: Advanced Deep Learning <i>Advanced Deep Learning</i>		8 ECTS/LP
Version 1.0.0 (seit WS18/19) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an dem Praxismodul verfügen die Studierenden über detailliertes und aktuelles Fachwissen auf dem Gebiet des maschinellen Lernens, können bedeutsame technische Entwicklungen identifizieren und können eine komplette Pipeline zur multimodalen Datenverarbeitung mit tiefen neuronalen Netzen implementieren. Sie können Problemstellungen und Ergebnisse des Gebiets präzise beschreiben, diskutieren und gelernte Konzepte und Methoden auf ähnliche Problemstellungen im maschinellen Lernen anwenden. Darüber hinaus analysieren die Studierenden weiterführende Konzepte, Methoden, Verfahren, Techniken und Technologien aus dem Gebiet des maschinellen Lernens, um sie in Forschungsprojekten anzuwenden, diese auf aktuelle industriennahe Aufgabenstellungen zu übertragen und dort aktiv mitzuarbeiten. Die Studierenden lernen wissenschaftlich anspruchsvolle Themenstellungen im Gebiet des maschinellen Lernens auf andere Forschungsfragen zu übertragen und darauf aufbauend ein komplexes Projekt in Gruppenarbeit auszuarbeiten. Außerdem verfügen sie über die Team- und Kommunikationsfähigkeit, um Problemstellungen auf dem Gebiet zu erörtern, Fragen und Zwischenergebnisse zu diskutieren, beschreiben und zu präsentieren. Zusätzlich sind die Studierenden in der Lage detaillierte Experimente durchzuführen und Ergebnisse zu beurteilen, vergleichen und auf Plausibilität zu prüfen.</p> <p>Schlüsselqualifikationen: Fortgeschrittene mathematisch-formale Methodik; Umsetzen fachlicher Lösungskonzepte in Programme und Modelle; Methoden zur Entwicklung größerer Softwaresysteme, Konstruktion von Abstraktionen und Architekturen; Fachübergreifende Kenntnisse; Systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten; Verstehen von Teamprozessen; Fertigkeit der Zusammenarbeit in Teams; Fähigkeit zur Leitung von Teams; Kennenlernen der Abläufe und Prozesse im Anwendungsumfeld der Informatik; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen; Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 20 Std. Vorlesung (Präsenzstudium) 40 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen: Grundkenntnisse in maschinellern Lernen und maschinellern Sehen (Grundstudiums-Vorlesungen "Multimedia Grundlagen 1" bzw. "Grundlagen der Signalverarbeitung und des Maschinellen Lernens", "Multimedia Grundlagen 2" sowie die Master-Vorlesung "Multimedia 2" bzw. "Machine Learning and Computer Vision")</p>		<p>ECTS/LP-Bedingungen: Bestehen der Portfolioprüfung</p>
<p>Angebotshäufigkeit: jedes Wintersemester</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 6</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	

Modulteile
Modulteil: Advanced Deep Learning (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2
Inhalte: <ul style="list-style-type: none">• Deep Learning in general• Deep Convolutional Neural Networks• Transfer Learning• Recurrent Neural Networks / LSTM Networks• Natural Language Processing• Multimodal Fusion (Vision+Language)• Application: Image Captioning
Zugeordnete Lehrveranstaltungen: Advanced Deep Learning (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i>
Modulteil: Advanced Deep Learning (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 4
Zugeordnete Lehrveranstaltungen: Übung zu Advanced Deep Learning (Übung) <i>*Veranstaltung wird online/digital abgehalten.*</i>
Prüfung Advanced Deep Learning Portfolioprüfung, Die Endnote setzt sich aus bewerteten Übungsblättern und einem bewerteten Teamprojekt zusammen.

Modul INF-0294: Speech Pathology <i>Speech Pathology</i>		5 ECTS/LP
Version 1.1.0 (seit WS18/19) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Knowledge: The students learn concepts relating to signal processing, speech production, phonetics, speech and language pathology, speech analysis, feature extraction, denoising and information reduction as exemplified through the analysis of automated voice pathology detection. They further gain insight into machine learning principles, with a particular focus on deep learning solutions, as is needed to diagnose a range of different voice pathologies. They will learn about different problems and solutions in the analysis of a variety of speech, relevant in the context of health care and wellbeing.</p> <p>Skills: The students will be able to choose appropriate algorithms of signal processing and machine intelligence, further develop these, design new solutions, and apply these to the task of voice pathology detection. They will know how to analyse and structure complex problems in the field, to employ suitable approaches to solve them, and to transfer knowledge to similar tasks. After participation in the course, they will be able to implement approaches and models into programs. Students will be able to assess developed systems in a scientific way. Important technical evolution and novelties in the fields of speech analysis and medical machine learning will be recognised by them.</p> <p>Competences: The students are able to characterise, judge on the quality and suitability, and design suited algorithmic solutions for intelligent signal analysis with a focus on voice pathology detection. They are further able to present and document results in a reasonable and meaningful way.</p> <p>Key skills: Formal methods; Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality awareness; Scientific working; Literature research.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>30 Std. Übung (Präsenzstudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen:</p> <p>Knowledge of basic mathematic lectures should be present.</p>		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Speech Pathology (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Englisch</p> <p>Angebotshäufigkeit: jedes Wintersemester</p> <p>SWS: 2</p>		

Inhalte:

The course "Pathological Speech" will give an introduction to models of speech production (e.g., source-filter models) with a focus on aspects that are relevant to pathologies and their recognition using automated methods of signal processing and machine learning. Moreover, students learn about robust feature extraction, modern methods of machine learning and machine intelligence, and implementation of such systems on devices

Topics: Speech production; Phonetics; Speech and language pathology; Signal processing; Natural language processing; Speech analysis; Feature extraction; Machine learning; Deep learning; Denoising; Information reduction; Healthcare.

Literatur:

- Björn Schuller, Anton Batliner: "Computational Paralinguistics: Emotion, Affect and Personality in Speech and Language Processing", Wiley, ISBN: 978-1119971368, 2013.
- Further literature is going to be announced during the lecture.

Zugeordnete Lehrveranstaltungen:

Speech Pathology (Vorlesung)

Veranstaltung wird online/digital abgehalten.

In this novel course, students will learn about human speech production and related pathologies. The course will introduce recent source-filter models and the principal concepts of signal processing, feature extraction, and information reduction. State-of-the-art machine learning techniques will be discussed as well as implications of practical applications of speech-related health care technologies.

Modulteil: Speech Pathology (Übung)

Lehrformen: Übung

Sprache: Englisch

Angebotshäufigkeit: jedes Wintersemester

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Speech Pathology (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Speech Pathology

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0296: Praktikum Interactive Machine Learning <i>Practical Module Interactive Machine Learning</i>		8 ECTS/LP
Version 1.0.0 (seit WS18/19) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierenden mit Grundkonzepten des interaktiven maschinellen Lernens vertraut. Sie sind in der Lage, fachliche Lösungskonzepte in Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Darüber hinaus erhalten sie einen Einblick in aktuelle Arbeiten des Forschungsgebiets. Weiterhin werden Kompetenzen in den Bereichen Team- und Kommunikationsfähigkeit sowie Selbstorganisation durch die Bearbeitung von Projektaufgaben und deren eigenständige Projektplanung vermittelt. Im Rahmen dieser Projektaufgaben werden Kenntnisse der Denkweise und Sprache des Forschungsgebietes durch die Diskussion und Präsentation der Projektergebnisse gefördert.</p> <p>Schlüsselqualifikationen: Umsetzen fachlicher Lösungskonzepte in Programme und Modelle; Fachspezifische Vertiefung; Fachübergreifende Kenntnisse; Fertigkeit, mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen anzustellen; Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen; Fertigkeit der Zusammenarbeit in Teams; Kenntnisse von praxisrelevanten Aufgabenstellungen; Kennenlernen der Abläufe und Prozesse im Anwendungsumfeld der Informatik; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Praktikum (Präsenzstudium) 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: Programmiererfahrung		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile
<p>Modulteil: Praktikum Interactive Machine Learning Lehrformen: Praktikum Sprache: Deutsch SWS: 6</p>
<p>Zugeordnete Lehrveranstaltungen: Praktikum Interactive Machine Learning (Praktikum) <i>*Veranstaltung wird online/digital abgehalten.*</i> Dieser Kurs fokussiert sich auf "interaktives Maschinelles Lernen" (iML). Wir definieren iML als Maschinelles Lernen bei dem der Mensch als zentraler Teil, interaktiv an verschiedenen Stellen in den Lernprozesses eingreift. Er überwacht dabei die Ergebnisse der Maschine und liefert Eingaben und Korrekturen um den Lernprozess zu verbessern. Während traditionelle ML Systeme menschliche Eingaben auf das Bereitstellen von Annotationen beschränken, beschäftigt sich iML damit Interaktionsdesign Richtlinien für ML Systeme zu erschaffen, und neue Methoden zu entwickeln menschliche Expertise in den ML Prozess einfließen zu lassen. Dadurch werden Systeme transparenter und nachvollziehbarer, was sich mit dem aktuell aufstrebenden Forschungstrend der "explainable AI" (XAI) überschneidet. Der Kurs besteht aus mehreren praktischen Hands-on Projekten und einer Einführung in verschiedene Themen rund um das Thema "interactive Machine Learning". Grundkenntnisse in Python sind erwünscht.</p>

... (weiter siehe Digicampus)

Prüfung

Praktikum Interactive Machine Learning

praktische Prüfung

Modul INF-0307: Modellbasierte Entwicklung und Analyse von Software Systemen <i>Model-Based Development and Analysis of Software Systems</i>		6 ECTS/LP
Version 1.1.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Modellbasierte Entwicklung und Analyse von Software Systemen befasst sich mit der Effizienzsteigerung in der Softwareherstellung durch Automatisierung und Wiederverwendung. Die Teilnehmer lernen in der Veranstaltung Methoden zur modellgetriebenen Entwicklung von Softwaresysteme anzuwenden und zu vergleichen. Sie entwickeln dabei vertiefte, fachspezifische Lösungskonzepte für MDSD. Sie können aktuelle Technologien und Standards für MDSD bewerten und analysieren deren Anwendbarkeit in praxisrelevanten Aufgabenstellungen. Die Teilnehmer bauen Fertigkeiten zur Analyse und Strukturierung komplexer Informatikproblemstellungen bei der Generierung von Infrastrukturcode, Subsysteme, Konfigurationen oder ganze Anwendungen aus Modellen auf. Dabei entwickeln sie Fertigkeiten zum logischen, analytischen und konzeptionellen Denken und können Problemlösungen systematisch entwickeln und beurteilen.</p> <p>Schlüsselqualifikation: Fachübergreifende Kenntnisse; Kompetenz zur Vernetzung unterschiedlicher Fachgebiete; Team- und Kommunikationsfähigkeit; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein; Kennenlernen der Abläufe und Prozesse im Anwendungsumfeld der Informatik; Formale quantitative Grundlagen kennen und verstehen; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen</p>		
<p>Arbeitsaufwand: Gesamt: 180 Std. 23 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 22 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 45 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: Frühere Veranstaltung "Modellgetriebene Softwareentwicklung" darf nicht belegt worden sein wegen Überschneidungen.		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 5	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Modellbasierte Entwicklung und Analyse von Software Systemen (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 3		
Inhalte: Modellbasierte Entwicklung und Analyse von Software Systemen befasst sich mit der Effizienzsteigerung in der Softwareherstellung durch Automatisierung und Wiederverwendung. Dabei werden Infrastrukturcode, Subsysteme, Konfigurationen oder ganze Anwendungen aus Modellen generiert.		

Literatur:

- Folien
- Pohl et al. Software Product Line Engineering: Foundations, Principles, and Techniques
- Kleppe et al: MDA explained
- Hitz et al: UML@Work
- weitere Literatur in der Vorlesung zu speziellen Themen

Modulteil: Modellbasierte Entwicklung und Analyse von Software Systemen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Modellbasierte Entwicklung und Analyse von Software Systemen

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0308: Software-intensive Systeme <i>Software-intensive Systems</i>		6 ECTS/LP
Version 1.1.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Die Studierenden sind in der Lage SW-Architekturen zu erstellen (K3), zu bewerten (K6) und zu dokumentieren. Dafür können sie fachliche Lösungskonzepte in Modelle überführen und kennen Methoden zur Entwicklung solcher Abstraktionen und Architekturen. Sie können Vor- und Nachteile von Entwurfsalternativen beschreiben (K4) und können diese im jeweiligen Anwendungszusammenhang bewerten (K6). Probleme können dabei eigenständig erkannt (K4) und Lösungen systematisch entworfen (K5) und realisiert (K3) werden. Weiterhin haben sie Fertigkeiten für die Analyse und Strukturierung der Problemstellungen in Unternehmensarchitekturen entwickelt und kennen die Konzepte und Vorgehensweisen für die Erstellung solcher Architekturen. Die Studierenden können praxisrelevanten Fragestellungen in Unternehmensarchitekturen benennen (K1). Sie können geeignete Methoden zur Architekturerstellung und -bewertung auswählen und sicher anwenden. Die Studenten kennen Modellierungssprachen und Patterns zur Erstellung von Software- und Unternehmensarchitekturen. Sie haben die Kompetenz bedeutende technische Entwicklungen zu erkennen.</p> <p>Schlüsselqualifikation: Kompetenz zur Vernetzung unterschiedlicher Fachgebiete; Team- und Kommunikationsfähigkeit; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Praxiserfahrung und Berufsbefähigung</p>		
<p>Arbeitsaufwand: Gesamt: 180 Std. 22 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 23 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 45 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Frühere Veranstaltung "Softwarearchitekturen und Enterprise Architecture Management" darf nicht belegt worden sein wegen Überschneidungen.		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 5	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Software-intensive Systeme (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 3		
Inhalte: Der Vorlesungsinhalt umfasst Patterns, Modellierungstechniken und die Evaluation von Softwarearchitekturen. Weiterhin wird auf den Bereich des Enterprise Architecture Managements eingegangen.		
Literatur: <ul style="list-style-type: none"> • Bass et al: Software Architecture in Practice • Clements et al: Documenting Software Architectures • Clements et al: Evaluation of Software Architectures • Kopetz: Real-Time Systems 		
Zugeordnete Lehrveranstaltungen:		

Software-intensive Systeme (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Inhalte: - Was sind SW-Architekturen? - Pattern und Muster für SW-Architekturen - Modellierung von SW-Architekturen - Evaluation von SW-Architekturen - Eingebettete Systeme: Definitionen, Anforderungsanalyse, Modellierung, Architektur - Software-Qualität: Definitionen und Standards, Funktionstest, Überdeckungsmaße, HiL-, Integrations- und Abnahmetests, Verifikation und Validierung, Architecture Design and Reliability - Enterprise Architecture Management: Methoden, Frameworks, Tools

Modulteil: Software-intensive Systeme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Software-intensive Systeme (Übung)

Veranstaltung wird online/digital abgehalten.

Der Übungsbetrieb soll die Inhalte der Vorlesung veranschaulichen und wird aus der gruppenweisen Vorstellung praktischer Beispiele von Architekturen und Frameworks bestehen. Die konkreten Themen folgen.

Prüfung

Software-intensive Systeme

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0315: Deep Learning <i>Deep Learning</i>		5 ECTS/LP
Version 1.2.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>The course Deep Learning covers the historical and formal fundamentals of Neural Networks, as well as the core principles of Machine Learning and data modelling.</p> <p>Upon completing the course, students will have the skills and knowledge to be able to choose suitable approaches and network architectures for specific tasks and know the pros and cons of design alternatives, as assessed in the respective application context. They will be able to apply and implement the discussed technical concepts in programs and systems. Furthermore, they will have the ability to analyse Deep Neural Network-based models and to design novel architectures and training methods.</p> <p>During the course, the participants will improve their skills in logical, analytical, and conceptual thinking. Students will gain the ability to make scientifically meaningful assessments in the field of machine learning and data science using appropriate methods. They will get used to the way of thinking and the language of relevant disciplines.</p> <p>Moreover, students will gain the ability to, convincingly, present their developed ideas and concepts. They will be able to apply their new knowledge to practical tasks and solve many real-life problems through the appropriate application of machine learning. They will also develop the competence to identify significant technical developments in the field.</p> <p>Key qualifications: analytical skills, data science cross-disciplinary knowledge, procedures and processes in creating practical systems, ability to present and document results in a comprehensible way, skill to solve problems under practical conditions, self-reflection, quality awareness, meticulousness, teamwork</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>30 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen:</p> <p>Knowledge of basic mathematic lectures should be present.</p>		
Angebotshäufigkeit: in der Regel mind. 1x pro Studienjahr	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Deep Learning (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Englisch		
SWS: 2		
<p>Inhalte:</p> <p>Perceptron, Feed-forward Neural Networks, Gradient-based Learning, Backpropagation, Recurrent Neural Networks, Convolutional Neural Networks, Autoencoders, Transfer Learning, Generative Adversarial Nets, Attention, Connectionist Temporal Classification, Data Preprocessing, Evaluation, Audio Classification, Object Detection, Natural Language Processing</p>		
<p>Literatur:</p> <p>Ian Goodfellow; Yoshua Bengio; Aaron Courville (2016). <i>Deep Learning</i>. Cambridge, Massachusetts: MIT Press.</p> <p>Further literature is going to be announced during the lecture.</p>		

Modulteil: Übung zu Deep Learning

Lehrformen: Übung

Sprache: Englisch

SWS: 2

Prüfung

Deep Learning

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0316: Machine Learning and Computer Vision <i>Machine Learning and Computer Vision</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul besitzen die Studierenden fortgeschrittene Kenntnisse des maschinellen Lernens (Entscheidungsbäume, neuronale Netze und tiefe neuronale Netze, Hypothesenevaluation, Instanz-basiertes Lernen, Bayessches Lernen, Lerntheorie), der Datenreduktion (z.B. Hauptkomponentenanalyse), der fortgeschrittenen Bildverarbeitung und des maschinellen Sehens und können diese anwenden. Sie sind in der Lage, wissenschaftlich komplexe Verfahren auf dem Gebiet der multimedialen Datenverarbeitung zu analysieren, zu verstehen und programmatisch umzusetzen, sowie die erlernten Prinzipien auf neue Fragestellungen geeignet anzuwenden. Sie entwickeln Fertigkeiten zum logischen, analytischen und konzeptionellen Denken im Bereich der digitalen Signalverarbeitung und multimedialen Datenverarbeitung.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Logik; umsetzen fachlicher Lösungskonzepte in Modelle; fachübergreifende Kenntnisse; Entwicklung und Umsetzung von Lösungsstrategien komplexer Problemstellungen; systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Machine Learning and Computer Vision (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 4		
<p>Inhalte: Die Vorlesung gibt einen guten Überblick über alle Aspekte des maschinellen Lernens und der maschinellen Extraktion von Informationen aus Multimediadaten (z.B. "Google Image Search", "Google Goggles"). Die erlernten Konzepte werden in den Übungen anhand von erfolgreichen Beispielen aus der Praxis ausprobiert, geübt, analysiert und bewertet. Zum Ende des Semesters werden fortgeschrittene Themen wie Objektdetektion und Objekterkennung von Gesichtern und Menschen behandelt. Die Inhalte der Vorlesung umfassen: Machine Learning (Decision Tree Learning, Artificial Neural Networks, Bayesian Learning, Discrete Adaboost), Data Reduction (Quantization (K-Means Clustering, Affinity Propagation), Dimensionality Reduction Techniques (PCA, NMF, Random Projection, MDS)) und Image Processing & Computer Vision (Salient Feature Points and Feature Descriptors, Object Detection (Face/Car/People Detection), Object Recognition (Face Recognition), Image Search with pLSA)</p>		

Literatur:

Literaturhinweise werden zu Beginn des Semesters bekanntgegeben.

Modulteil: Machine Learning and Computer Vision (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Machine Learning and Computer Vision (Klausur)

Klausur / Prüfungsdauer: 120 Minuten

Beschreibung:

Die Prüfung kann jedes Semester in der Prüfungszeit abgelegt werden.

Modul INF-0318: Praktikum Simulation von autonomen Fahrzeugen (Vertiefung) <i>Practice Module on Simulation of Autonomous Vehicles</i>		8 ECTS/LP
Version 1.0.0 (seit WS19/20) Modulverantwortliche/r: Prof. Dr. Lars Mikelsons		
Lernziele/Kompetenzen: Fachbezogene Kompetenzen: Nach der Teilnahme am Praktikum verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet der Simulation von autonomen Fahrzeugen und verfügen dort über tiefere Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Kleingruppen entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und bestehende Konzepte umzusetzen. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.		
Schlüsselkompetenzen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Validierung von Simulationsergebnissen und Softwaremodulen; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik;		
Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Praktikum (Präsenzstudium) 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)		
Voraussetzungen: Gute Programmierkenntnisse Hilfreich: Python, C++, ROS, Game Engines		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Simulation von autonomen Fahrzeugen (Vertiefung) Lehrformen: Praktikum Sprache: Deutsch / Englisch SWS: 6 ECTS/LP: 8.0		
Inhalte: Im Praktikum bearbeiten die Studierenden in Kleingruppen anwendungsorientierte Aufgaben zur Simulation von autonomen Fahrzeugen. Dabei dürfen Sie einzelne Softwaremodule eines virtuellen Fahrzeugs selbst implementieren und validieren.		
Zugeordnete Lehrveranstaltungen:		

Praktikum Simulation von autonomen Fahrzeugen (Vertiefung) (Praktikum)

Veranstaltung wird als Hybrid/gemischt abgehalten.

Anmeldung bis 30. Oktober 2020. Teilnehmer werden über Zulassung zum Praktikum dann informiert.

Prüfung

Praktikum Simulation von autonomen Fahrzeugen

Portfolioprüfung

Modul INF-0353: Praktikum Programmieren in der biomedizinischen Informatik (Master) <i>Practical Module Biomedical Programming (Master)</i>		6 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Frank Kramer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Praktikum Programmieren in der Biomed. Informatik verstehen die Studierenden praxisnahe Problemstellungen höherer Komplexität im Bereich Softwareentwicklung und Auswertungen in den Anwendungsbereichen der biomedizinischen Informatik. Die Studierenden erlangen tiefere fachspezifische als auch fächerübergreifende Kenntnisse und Fähigkeiten, beispielsweise Bioinformatik, Medizininformatik und Statistik. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren, zu präsentieren und verständlich zu dokumentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Projektmanagementfähigkeiten</p>		
<p>Bemerkung: Wenn Sie bereits das Modul "INF-0326: Praktikum Programmieren in der biomedizinischen Informatik (Vertiefung)" gehört haben, ist eine erneute Einbringung nicht möglich!</p>		
<p>Arbeitsaufwand: Gesamt: 180 Std. 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 90 Std. Praktikum (Präsenzstudium) 30 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Moduleil: Praktikum Programmieren in der biomedizinischen Informatik (Master) Lehrformen: Praktikum Sprache: Deutsch SWS: 6</p>		

Inhalte:

Die Teilnehmer des Bio-/Medizininformatik Programmierpraktikums erlernen, wie man kleinere Anwendungs- und Infrastrukturprogramme im biologischen und medizinischen Kontext implementiert.

Ziel des Praktikums ist es, dass die Studierenden diverse kleinere Programme in Python zu medizinischen Fragestellungen entwickeln.

Die Studierenden erwerben anhand von täglichen Arbeitsaufgaben Grundkenntnisse über:

- Python Programmierung
- Nutzung von Bio-/Medizinischen Datenbanken
- Datentransformation im biologischen und medizinischen Kontext
- Krankheitsforschung in der Bio-/Medizininformatik
- Hochdurchsatzdaten Analyse

Das Praktikum wird als 2-wöchige Blockveranstaltung in den Semesterferien angeboten, besteht aus einer kurzen täglichen Einführung zu den aktuellen Arbeitsaufgaben und der anschließenden selbstständigen Implementierung durch die Studierenden.

Während des selbstständigen Arbeitens wird zu festgelegten Zeiten einer Betreuer für Hilfestellungen und Fragen anwesend sein.

Während des Semesters wird es ein einmaliges Treffen aus organisatorischen Gründen geben.

Prüfung

Praktikum Programmieren in der biomedizinischen Informatik (Master)

praktische Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0367: Advanced Machine Learning and Computer Vision		5 ECTS/LP
Version 1.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul besitzen die Studierenden vertiefte fortgeschrittene Kenntnisse des maschinellen Lernens (Support Vector Machines und tiefe neuronale Netze und deren Grundbausteine) und des maschinellen Sehens (tiefe neuronale Netzarchitekturen und Systeme) und können diese anwenden. Sie sind in der Lage, wissenschaftlich komplexe Verfahren auf dem Gebiet der Bild-, Text-, Video- und Signalverarbeitung zu analysieren, zu verstehen und programmatisch umzusetzen, sowie die erlernten Prinzipien auf neue Fragestellungen geeignet anzuwenden. Sie entwickeln Fertigkeiten zum logischen, analytischen und konzeptionellen Denken im Bereich des maschinellen Lernens und Sehens.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Logik; kritisches Lesen und Analysieren von wissenschaftlichen Veröffentlichungen; Umsetzen fachlicher Lösungskonzepte in Modelle; fachübergreifende Kenntnisse; Entwicklung und Umsetzung von Lösungsstrategien komplexer Problemstellungen; systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
<p>Voraussetzungen: Kenntnisse in maschinellern Lernen und maschinellern Sehen (Master-Vorlesung INF-0092 "Multimedia II" bzw. INF-0316 "Machine Learning and Computer Vision")</p>		<p>ECTS/LP-Bedingungen: Bestehen der Modulprüfung</p>
<p>Angebotshäufigkeit: jedes Wintersemester</p>	<p>Empfohlenes Fachsemester: ab dem 2.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 4</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p>		
<p>Modulteil: Advanced Machine Learning and Computer Vision (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		
<p>Inhalte: Die Vorlesung gibt einen vertieften Einblick in alle Aspekte des maschinellen Lernens und des maschinellen Sehens. Die erlernten Konzepte werden in den Übungen anhand von erfolgreichen Beispielen aus der Praxis ausprobiert, geübt, analysiert und bewertet. Die Inhalte der Vorlesung umfassen: Support Vector Machines, Grundbausteine von tiefen Neuronalen Netzen (Layerstrukturen, Normalisierung, Attention-Mechanismen), sowie aktuelle Referenzarchitekturen und -systeme für Bild-, Text-, Videoverarbeitung und deren Kombination mit weiteren Sensorsignalen.</p>		
<p>Literatur: Literaturhinweise werden zu Beginn des Semesters bekanntgegeben.</p>		
<p>Zugeordnete Lehrveranstaltungen:</p>		

Advanced Machine Learning and Computer Vision (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Modulteil: Advanced Machine Learning and Computer Vision (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Advanced Machine Learning and Computer Vision (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Advanced Machine Learning and Computer Vision (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Beschreibung:

Die Prüfung kann jedes Semester in der Prüfungszeit abgelegt werden.

Modul ZCS-2000: Softskills	2 ECTS/LP
Version 1.0.0 (seit SoSe16) Modulverantwortliche/r: Claudia Lange-Hetmann	
Inhalte: Detailbeschreibungen zu Kursen und Anmeldeverfahren befinden sich auf https://www.uni-augsburg.de/de/studium/zusatzqualifikationen/profilbildung/#Anker_skK bzw. im digicampus.	
Lernziele/Kompetenzen: Die Studierenden erwerben in diesem Modul - je nach Kurswahl - entweder kommunikative, soziale oder methodische Fähigkeiten, die unerlässlich für ihre künftige Berufsfähigkeit sind, denn diese fordert eine überzeugende Persönlichkeit des Einzelnen und eine einwandfreie und zielgerichtete Interaktion im Team. Daher wird bei der Auswahl empfohlen, einen Kurs aus einem der drei Kompetenzgebiete zu wählen, die zur Stärkung der eigenen Persönlichkeit sinnvoll und wichtig sind. Zudem bildet die interdisziplinäre Zusammensetzung der Teilnehmer aus unterschiedlichen Fachrichtungen den typischen Wirkungskreis im späteren Arbeitsumfeld ab. Die Studierenden werden - abhängig von der Kursthemenwahl - - neben dem Erwerb der Fertigkeit der verständlichen, sicheren und überzeugenden Darbietung von Ideen, Konzepten und Ergebnissen bzw. dem Verständnis der psychologischen Grundlagen von Dialogen und Verhandlungen dieses Wissen anwenden können, um Interesse, Verständlichkeit und Sympathie zu erzeugen und zielorientiert zu präsentieren bzw. zu argumentieren. - die Kommunikations-, Dialog- und Teamprozesse in Bezug auf Motivation, Effektivität und kennen die Entstehung, Dynamik, Lösung und Prävention von Konflikten verstehen und können Moderationstechniken und ihre Fertigkeit zur Selbstreflexion anwenden, sie beherrschen die Regeln bei der Teamarbeit, bei Besprechungen bis hin zur Führung von Teams oder kennen den Nutzen von gesellschaftlichem Engagement für sich und die Gesellschaft. - grundlegende Konzepte des Projektmanagements (u.a. Entwurf von strategischen Projektstrukturplänen, Analyse der Projektumwelt/-risiken, Projektcontrolling) verstehen und können die Grundlagen der Motivationspsychologie und zentrale Führungstechniken zur Erreichung des Projekterfolgs anwenden. Oder sie können grundlegende Strategien und Methoden für die Entwicklung und Absicherung einer Unternehmensführung anwenden, sie kennen Marketing- u. Vertriebsstrategien, bewerten deren Erfolgsaussichten und haben Kenntnisse in Personal- und Finanzmanagement. Sie verstehen Probleme zu analysieren und können konstruktiv im Team eine Lösung erarbeiten und kommunizieren oder vertiefen Teilaspekte wie u.a. Kreativität, Innovationsfähigkeit mit innovativen Methoden. Besonderer Wert wird - je nach Kurs - auf die Weiterentwicklung der eigenen Präsentations- und Kommunikationsfähigkeit, der Teamkompetenz sowie die Anwendung des Methodenwissens und die Erreichung realistischer Ziele gelegt. Die interdisziplinäre Herangehensweise an eine Problemstellung wird durch die interdisziplinäre Zusammensetzung der Kleingruppen in den Kursen trainiert, durch praktische Übungen in den Kursen gefestigt und durch Selbstreflexion und Feedbackmethoden verinnerlicht.	
Bemerkung: <u>Anmeldungsfrist:</u> Für die Teilnahme an den Kursen ist eine Anmeldung über digicampus erforderlich. <u>Anmeldephase:</u> Jan (für das folgende SS) bzw. Juli (für das folgende WS). Die Kurse finden größtenteils ab März bis letzten Sa* im April (SS) bzw. ab Sep. bis letzten Sa* im Okt. *vor Vorlesungsbeginn statt sowie einige in der Vorlesungszeit (Fr/Sa). Die Kurse haben eine limitierte Teilnehmerzahl pro Semester.	
Arbeitsaufwand: Gesamt: 60 Std. 20 Std. Seminar (Präsenzstudium) 10 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 20 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 10 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)	

Voraussetzungen: keine		ECTS/LP-Bedingungen: aktive Übungsteilnahme im Kurs
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 2.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	

Moduleile	
Moduleil: Softskills	
Lehrformen: Kurs	
Sprache: Deutsch / Englisch	
SWS: 2	
ECTS/LP: 2.0	
Inhalte:	
<p>Zur Auswahl stehen nachfolgende Kurse/Themen:</p> <p><u>(1) Kommunikationskompetenz</u></p> <ul style="list-style-type: none"> - Kommunikationstraining - Rhetorik - Präsentation (1 bz. 2-tägige Kurse) - Strategische Gesprächsführung - Feedback geben (1 Tag - daher 2. Kurs notwendig!) - Kommunikation in Projekten (1 Tag - daher 2. Kurs notwendig!) - Communication in engl. (1 Tag - daher 2. Kurs notwendig!) <p><u>(2) Sozialkompetenz</u></p> <ul style="list-style-type: none"> - Konfliktmanagement - Emotionale Intelligenz - Moderation & Teamleitung - Führungskompetenzen entwickeln - Gesellschaftliches Engagement <p><u>(3) Methodenkompetenz</u></p> <ul style="list-style-type: none"> - Zeit-/Selbstmanagement - Changemanagement - Meetings erfolgreich moderieren (1 Tag - daher 2. Kurs notwendig!) - Innovationen entwickeln - Design Thinking - Projektmanagement (dt./ engl.) - Unternehmerisches Denken <p>Weiterehin können auch Kompakt-Kurse gewählt werden, bei denen die Teilnehmer o.g. Fähigkeiten erlernen und eine Projektaufgabe im Team bearbeiten. Der höhere Zeitaufwand wird mit mehr Erfahrung honoriert.</p> <ul style="list-style-type: none"> - Kompaktkurs Märkte für Menschen - Kompaktkurs Projekte real durchführen - Kompaktkurs Projekt "Expedition" - ACE Startup-Challenge <p>Detailbeschreibungen zu allen Kursen sowie die konkreten Kursthemen und Termine pro Semester im digicampus.</p>	
Lehr-/Lernmethoden:	
<p>Vortrag, Diskussion, Übungen, Praxisbeispiele, event. Projektarbeit unter Verwendung von multimedialen Techniken (Beamer, Flipchart, Pinwand)</p>	
Literatur:	
<p>wird im Kurs bz. in die Kursbeschreibungen angeben bzw. vorab kommuniziert.</p>	

Zugeordnete Lehrveranstaltungen:

Kompaktkurs - Projekt "Expedition" (Kurs)

Veranstaltung wird in Präsenz abgehalten.

Projektarbeit wird sowohl im Studium als auch im Beruf gefordert und verlangt neben fachlichem und methodischem Knowhow auch Fähigkeiten wie Kommunikationsgeschick und Verantwortlichkeitsgefühl. Lernen Sie Projekte effizient und geordnet sowie mit Freude durchzuführen, die Teammitglieder zu motivieren und nach ihren Fähigkeiten einzusetzen, gemeinsam auf ein sinnvolles Ziel zuzusteuern und am Ende das Ergebnis gemeinsam entsprechend in Szene zu setzen. In diesem Intensivkurs, werden Sie mit fachlicher Anleitung ein mehrtägiges Projekt aus dem realen Arbeitsalltag des Projektpartners durchführen und dabei wertvolle Erfahrungen sammeln. Begleitend werden Sie Ihre Potentiale und die des Teams entdecken und stärken lernen, denn unser Projektpartner Outward Bound Germany bietet neben der Projektaufgabe auch ein interessantes Kurssetting. Lerninhalte: • Aufbau von Projektteams und die Verschiedenen Rollen /Aufgaben • Projektplanung und zielspezifische Umsetzung • Projektpräsentation und -absc
... (weiter siehe Digicampus)

Kompaktkurs - Projekte real durchführen (Kurs)

Veranstaltung wird online/digital abgehalten.

Projektarbeit und Team-Work brauchen Sie sowohl im Studium als auch im späteren Berufsleben. Gleichmaßen ist es heute im Arbeitsalltag unumgänglich, mit innovativen und agilen Methoden an komplexe Fragestellungen heranzugehen. Dieser Intensivkurs stellt Sie auf die Probe! Arbeiten Sie im Team eine Woche an einem konkreten Projekte. Lernen Sie am Vormittag in intensiven Online-Seminaren ausgewählte Inhalte aus Teamführung, agilem Projektmanagement, Design Thinking, Business Modelling und sicherem Präsentieren. Am Nachmittag arbeiten Sie in Ihrem Team remote an einem Projekt. Zusätzlich erarbeiten Sie sich im Selbststudium auf einer Lernplattform weitere Inhalte. Jeder Tag wird mit einem Review der Ergebnisse beendet. Begleitend werden Sie Ihre Potentiale und die des Teams entdecken und stärken lernen, denn unser Projektpartner Steinbeis IFEM bietet neben der Projektaufgabe eine langjährige Erfahrung bei der Durchführung von Trainings in allen Ebenen diverser Unternehmen und Organisati
... (weiter siehe Digicampus)

Kompaktkurs Märkte für Menschen - Veränderungen gestaltet (Kurs)

Veranstaltung wird als Hybrid/gemischt abgehalten.

Inhalte: In diesem Kompaktkurs werden Inhalte von gesellschaftlicher Relevanz interdisziplinär erschlossen. Hierbei werden Themenbereiche wie Ernährungsgewohnheiten vs. globale Ernährungssicherheit, (Finanz-)Märkte vs. Gerechtigkeit kontrovers diskutiert. Jeder dieser Themenbereiche wird zunächst mit Hilfe wissenschaftlicher Inputs interdisziplinär erschlossen. Daran anschließend erarbeitet sich jede Kleingruppe fundiertes Wissen einem Themenbereich, setzt sich differenziert mit kontroversen Perspektiven auseinander und präsentiert die Ergebnisse anschließend kontrovers in einer „Talk-Show“ (Podiumsdiskussion). Die Studierenden erarbeiten in Projektgruppen selbst konkrete Vorschläge und Projekte zur Gestaltung einer besseren Zukunft im Rahmen der Themenkomplexe und präsentieren diese zum Abschluss des Seminars. Parallel zu dieser inhaltlichen Arbeit werden methodische Kenntnisse und Hintergrundinformationen vermittelt, um die TeilnehmerInnen dazu zu befähigen, selbst die Zukunft aktiv
... (weiter siehe Digicampus)

Kompaktkurs: ACE Startup Challenge (Kurs)

Veranstaltung wird online/digital abgehalten.

Im innovativen, interdisziplinaren Seminarkonzept bekommen die Studierenden einen Startup Real-Case, an dem Sie ihr ganzes unternehmerisches Talent unter Beweis stellen dürfen. Dabei werden sie von Experten und Startup-Mentoren begleitet; Fachvorträge und die Möglichkeit mit der Augsburger Startup Szene auf Tuchfühlung zu gehen, runden das Seminarangebot ab. Nach der Teilnahme der Modulveranstaltung sind die Studierenden in der Lage, wesentliche Themen- und Problemstellungen rund um Startups und Unternehmensgründungen zu kennen, zu analysieren und geeignete Strategie abzuleiten. Insbesondere erhalten die Studierenden ein tiefes Verständnis für Geschäftsmodelle, Fragen des Pricings und Marketings sowie rechtliche Fragestellungen. Die Studierenden erlernen Kenntnisse zur Anwendung der wichtigsten Methoden und Konzepte zum Business Modeling, der

Budgetierung, Marktanalysen und Unternehmensfinanzierung. Die Studierenden können nach dem Studium des Moduls unternehmerische Problemstellungen
... (weiter siehe Digicampus)

Kurs Business Knigge erleben (Kurs)

Veranstaltung wird online/digital abgehalten.

Warum ist es auch heute noch wichtig, angemessene Umgangsformen und ein passendes Auftreten bei öffentlichen Anlässen zu kennen und anzuwenden? Sind „Knigge und Etikette“ noch aktuell? In Theorie und Praxis werden wir Themen wie passende Begrüßungsformen und Verabschiedungen, Small Talk, Umgangsformen, authentisches Auftreten und vieles mehr gemeinsam bearbeiten. Wir werden gemeinsam diskutieren, spielerisch die Fragestellungen erarbeiten und uns austauschen. Gezeit werden Sie angemessene Umgangsformen im Beruf und Gesellschaft kennen lernen und ein Bewusstsein entwickeln für persönliche Entscheidung, um beim professionellen Auftreten authentisch zu bleiben. Kursinhalte: o Begrüßung und Empfang – der erste Eindruck zählt o Verbale und non-verbale Kommunikation – small talk und mehr o Kleider machen Leute – wie kleide ich mich richtig im Beruf? o Mein persönlicher Stil – was ist authentisch und passt zu mir? o Persönlicher Kontakt - Umgangsformen (am Tisch) o Professioneller Kontakt pe
... (weiter siehe Digicampus)

Kurs Changemanagement (Kurs)

Veranstaltung wird online/digital abgehalten.

Veränderungen effizient gestalten, Widerstände positiv wandeln. Als Fach- und Führungskraft ist es Ihre Aufgabe, Veränderungen in Ihrem Unternehmen aktiv zu gestalten und erfolgreich umzusetzen. Wie können Unternehmen die Herausforderungen einer sich ständig wandelnden Welt begegnen um ihr Überleben zu sichern? Möchten Sie Veränderungen erfolgreich zum Ziel bringen und Ursachen von Widerständen verstehen? Wollen Sie wissen, wie Sie in schwierigen Situationen schneller Lösungen finden und ein Team firmieren, das effizient zusammenarbeitet? Change Management kann Ihnen dabei helfen, den notwendigen Wandel systematisch, dh. bewusst zu gestalten. Veränderungen gehen dabei oft mit Ängsten und einer Abwehrhaltung der Menschen einher. Lernen Sie in diesem Kurs Veränderungen erfolgreich zu bewältigen und mit Widerständen umzugehen. Erfahren Sie sehr anschaulich, wie Veränderungsprozesse gesteuert werden können, Widerstände erst gar nicht entstehen und falls doch zielorientiert aufgelöst werden
... (weiter siehe Digicampus)

Kurs Emotionale Intelligenz (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Die Forschung zur Emotionalen Intelligenz zeigt, dass Schlüsselkompetenzen neben fachlichen und funktionalen Kompetenzen für den Studienerfolg und die berufliche Karriere von ausschlaggebender Bedeutung sind (Goleman, 2017). Der Kurs „Emotionale Intelligenz“ baut auf dem gleichnamigen Konzept von Daniel Goleman (2011) auf und gibt einen allgemeinen Überblick über seine Arbeiten und Erkenntnisse. Nach Goleman sind fünf Kompetenzen für den Schul-, Studien- und Berufserfolg sowie generell für eine erfolgreiche Lebensführung verantwortlich: Selbstreflexion, Selbstbeherrschung, Selbstmotivation, Empathie und Soziale Kompetenz. Im Kurs werden diese erfolgsrelevanten Kompetenzen durch geeignete Methoden und didaktische Ansätze bei den Kursteilnehmern konsequent (weiter) entwickelt. Auf diese Weise erwerben die Teilnehmer nicht nur relevantes Wissen über die Emotionale Intelligenz, sondern auch praktische Fähigkeiten zu ihrer wirksamen Anwendung. Weitere zentrale Inhalte des Kurses sind: - Die
... (weiter siehe Digicampus)

Kurs Emotionale Intelligenz (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Die Forschung zur Emotionalen Intelligenz zeigt, dass Schlüsselkompetenzen neben fachlichen und funktionalen Kompetenzen für den Studienerfolg und die berufliche Karriere von ausschlaggebender Bedeutung sind (Goleman, 2017). Der Kurs „Emotionale Intelligenz“ baut auf dem gleichnamigen Konzept von Daniel Goleman (2011) auf und gibt einen allgemeinen Überblick über seine Arbeiten und Erkenntnisse. Nach Goleman sind fünf Kompetenzen für den Schul-, Studien- und Berufserfolg sowie generell für eine erfolgreiche Lebensführung verantwortlich: Selbstreflexion, Selbstbeherrschung, Selbstmotivation, Empathie und Soziale Kompetenz. Im Kurs werden diese erfolgsrelevanten Kompetenzen durch geeignete Methoden und didaktische Ansätze bei den

Kursteilnehmern konsequent (weiter) entwickelt. Auf diese Weise erwerben die Teilnehmer nicht nur relevantes Wissen über die Emotionale Intelligenz, sondern auch praktische Fähigkeiten zu ihrer wirksamen Anwendung. Weitere zentrale Inhalte des Kurses sind: - Die ... (weiter siehe Digicampus)

Kurs Emotionale Intelligenz (Option 3) (Kurs)

Veranstaltung wird online/digital abgehalten.

Die Forschung zur Emotionalen Intelligenz zeigt, dass Schlüsselkompetenzen neben fachlichen und funktionalen Kompetenzen für den Studienerfolg und die berufliche Karriere von ausschlaggebender Bedeutung sind (Goleman, 2017). Der Kurs „Emotionale Intelligenz“ baut auf dem gleichnamigen Konzept von Daniel Goleman (2011) auf und gibt einen allgemeinen Überblick über seine Arbeiten und Erkenntnisse. Nach Goleman sind fünf Kompetenzen für den Schul-, Studien- und Berufserfolg sowie generell für eine erfolgreiche Lebensführung verantwortlich: Selbstreflexion, Selbstbeherrschung, Selbstmotivation, Empathie und Soziale Kompetenz. Im Kurs werden diese erfolgsrelevanten Kompetenzen durch geeignete Methoden und didaktische Ansätze bei den Kursteilnehmern konsequent (weiter) entwickelt. Auf diese Weise erwerben die Teilnehmer nicht nur relevantes Wissen über die Emotionale Intelligenz, sondern auch praktische Fähigkeiten zu ihrer wirksamen Anwendung. Weitere zentrale Inhalte des Kurses sind: - Die ... (weiter siehe Digicampus)

Kurs Innovationen entwickeln (Kurs)

Veranstaltung wird online/digital abgehalten.

Großartige Innovationen wie das iPhone, die VR-Brille Oculus Rift oder der Tesla Model S sind der Traum eines jeden Unternehmers. Doch wie kommt man auf solche Ideen und wie wird daraus ein erfolgreiches Produkt? Wie das Scheitern des Alpha Patentfonds zeigt, sind Ideen und selbst erteilte Patente nicht zwingend durchstartend. Eine Idee wird dann zur Innovation, wenn es gelingt, sie zur Realisierungsreife zu bringen und Vorgesetzte, Partner und Geldgeber davon zu begeistern, die Marktchancen realistisch eingeschätzt werden, die technische Machbarkeit und vor allem der wirtschaftliche Nutzen gegeben sind sowie das Projekt- oder Gründerteam und das richtige Timing im Marktumfeld stimmt. Kursinhalte: • Innovationsentwicklung und –identifikation mithilfe von - Kreativitätstechniken (Brainstorming, Rollenwechsel, Bionik, Was-wäre-wenn-Methode, Denkhüte von De Bono) - Trendadaption (360° Scanning, Trendanalyse) - Design Thinking Methode • Praktische Anwendung als Innovationsprozess in der Gr ... (weiter siehe Digicampus)

Kurs Kommunikationstraining (Kurs)

Veranstaltung wird online/digital abgehalten.

In diesem Online-Seminar lernen Sie durch authentische wertschätzende Kommunikation gerade in herausfordernden Zeiten zu begeistern, Emotionen zu wecken und erfolgreich einzusetzen. Erleben Sie, wie Sie professionell strukturierte Gespräche effektiv, klar und überzeugend führen. Die Inhalte des praxisorientierten Online Trainings sind sofort in Ihrem Studium und Alltag erfolgreich einsetzbar! Inhalte: • Kommunikation gerade in schwierigen Zeiten zielorientiert ausrichten • Überzeugende Fragetechnik – Wer fragt, der führt • In 60 Sekunden begeistern mit strukturierter Elevator Pitch • Menschen- und Kundentypen besser einschätzen • Gespräche positiv mit Anerkennung und Wertschätzung steuern • Tipps und Werkzeuge für mehr Selbstsicherheit in der Kommunikation Methodik: Aktives praxisorientiertes Live Online Training mit Partnerübungen im virtuellen Klassenraum und Einzelübungen. Leistungsnachweis: Anwesenheit an allen synchronen Terminen, Bearbeitung aller Arbeitsaufträge. Dozentin: Micha ... (weiter siehe Digicampus)

Kurs Konfliktmanagement (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Konflikte gehören zum Alltag wie auch zum Berufsleben. Konflikte sind allgegenwärtig. Umso wichtiger ist es zu wissen, wie man Konflikte konstruktiv löst und für beide Seiten gewinnbringend in Positives verwandeln kann. Lernen Sie sich und Ihre Mitmenschen besser kennen. Wir erarbeiten mit Ihnen zusammen die Techniken, um auch in schwierigen Situationen gelassen und zielorientiert zu agieren. Lerninhalte: • Was ist ein Konflikt? • Wie entsteht er? • Wie löst man Konflikte konstruktiv? • Nullsummenspiel vs. Win-Win Situation • Killerphrasen

entlarven Methoden: Theorie in Kombination mit konkreten Beispielen aus der Praxis, praktische Übungen und viele unterschiedliche Tools Dozentin: Natalie Raess-Beuchle, Coraessco Coaching & Consulting

Kurs Konfliktmanagement (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Konflikte sind allgegenwärtig. Sie gehören sowohl zum privaten Alltag wie auch zum Berufsleben. Daher ist es wichtig zu wissen, wie man Konflikte konstruktiv löst und für alle Beteiligten gewinnbringend in positive Veränderungen verwandeln kann. Die Teilnehmer erarbeiten gemeinsam Vorgehensweisen, um auch in schwierigen Konfliktsituationen gelassen und zielorientiert zu agieren. Vor dem Hintergrund der Prinzipien gewaltfreier Kommunikation (Rosenberg, 2016) lernen die Teilnehmer, welche Gesprächshaltungen, -formen und -techniken zur Konflikteskalation bzw. zur De-Eskalation beitragen. Mittels der gewaltfreien Bewegungskunst „Aikido“ können die Teilnehmer in Körperübungen erleben, wie man Angriffsenergien aufnehmen und umlenken kann. Aikido zeigt, wie man Konflikte ohne Konfrontation oder Widerstand behandelt. Anschließend wird der Bewegungsablauf des Aikido durch Analogiebildung auf das Führen von Konfliktgesprächen übertragen. Das entsprechende Gesprächsmodell ist Aikikom (= Kommunika
... (weiter siehe Digicampus)

Kurs Meetings erfolgreich moderieren (Kurs)

Veranstaltung wird online/digital abgehalten.

Eventuell bereits im Studium und sicher im Berufsleben sind Besprechungen ständige Begleiter. Bestens vorbereitete und erfolgreich durchgeführte Besprechungen sind dennoch eine Seltenheit. Dabei kann man gutes Besprechungsmanagement ganz einfach trainieren und mit dieser Kompetenz in Zukunft glänzen. Inhalte:
• Welche Besprechungsarten gibt es? • Wie bereite ich eine Besprechung professionell vor? • Wie leite ich zielführend durch die verschiedenen Besprechungsphasen? • Wie bringe ich meine Botschaft überzeugend und zielgruppengerecht an den Mann/die Frau? • Wie nutze ich dabei Visualisierungen? • Wie bringe ich Besprechungen zu einem verbindlichen Abschluss? • Wie gehe ich mit unterschiedlichen Besprechungssituationen um? Methoden: Theorie-Input, Tipps aus der Praxis, Fallbeispiele und Übungen, Reflexion, ausführliches Feedback, Kollegiale Beratung, Diskussion und Einzel-Coaching. Dozentin: Nina Turani, Seniorreferentin Personal- und Führungskräfteentwicklung, DB Fernverkehr AG
... (weiter siehe Digicampus)

Kurs Project management (english) (Kurs)

Veranstaltung wird online/digital abgehalten.

Projects are important at all company aspects and resorts. Essential for success is that all project members know and accept the project goals, plan and their own tasks as well as an efficient project coordination and controlling. Therefore the course trains fundamental concepts of modern project management. Lerninhalte: • Basics of Project management • Project definition • Project organisation • Project communication • Project planning • Project calculation • Project risk management • Project controlling • Project closing • Project documentation Methoden: teacher presentation (in parts), working on a business case in small groups, presenting the business case solution per team at the end, detailed feedback from all attendencies and course leader Dozent: Prof. Dr. Matthias Menter (Jun.-Prof.), Chair of Business Dynamics, Innovation and Economic Change, Friedrich Schiller University Jena
... (weiter siehe Digicampus)

Kurs Projektmanagement (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Projekte stellen eine immer bedeutsamer werdende Form zur Unternehmensführung dar. Maßgeblich für deren Erfolg sind effiziente Koordinierung sowie zielfördernde Beiträge seitens der Projektbeteiligten. Daher vermittelt dieser Kurs grundlegende Konzepte modernen Projektmanagements. Lerninhalte: • Grundlagen des Projektmanagements • Projekte auswählen und Projektziele definieren • Projekte planen und effizient kontrollen • Projektstrukturpläne entwerfen und Meilensteine setzen • Projekte kosteneffizient kalkulieren • Projektrisiken erkennen und managen • Projekte zielorientiert dokumentieren • Projekte erfolgreich abschließen Methoden: Vortrag durch Referenten, Fortlaufende Bearbeitung einer Fallstudie in Kleingruppen, Abschlusspräsentation der jeweiligen Fallstudie durch die Kursteilnehmer, Ausführliches Feedback durch Kursteilnehmer und -leiter Dozent: Prof. Dr. Matthias Menter (Jun.-Prof.), Lehrstuhl für Unternehmensentwicklung, Innovation und wirtschaftlichen Wandel, Friedrich-Schill

... (weiter siehe Digicampus)

Kurs Projektmanagement (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Projekte stellen eine immer bedeutsamer werdende Form zur Unternehmensführung dar. Maßgeblich für deren Erfolg sind effiziente Koordinierung sowie zielfördernde Beiträge seitens der Projektbeteiligten. Daher vermittelt dieser Kurs grundlegende Konzepte modernen Projektmanagements. Lerninhalte: • Grundlagen des Projektmanagements • Projekte auswählen und Projektziele definieren • Projekte planen und effizient kontrollieren • Projektstrukturpläne entwerfen und Meilensteine setzen • Projekte kosteneffizient kalkulieren • Projektrisiken erkennen und managen • Projekte zielorientiert dokumentieren • Projekte erfolgreich abschließen Methoden: Vortrag durch Referenten, Fortlaufende Bearbeitung einer Fallstudie in Kleingruppen, Abschlusspräsentation der jeweiligen Fallstudie durch die Kursteilnehmer, Ausführliches Feedback durch Kursteilnehmer und -leiter Dozent: Prof. Dr. Matthias Menter (Jun.-Prof.), Lehrstuhl für Unternehmensentwicklung, Innovation und wirtschaftlichen Wandel, Friedrich-Schill

... (weiter siehe Digicampus)

Kurs Projektmanagement (Option 3) (Kurs)

Veranstaltung wird online/digital abgehalten.

Projekte stellen eine immer bedeutsamer werdende Form zur Unternehmensführung dar. Maßgeblich für deren Erfolg sind effiziente Koordinierung sowie zielfördernde Beiträge seitens der Projektbeteiligten. Daher vermittelt dieser Kurs grundlegende Konzepte modernen Projektmanagements. Lerninhalte: • Projektanforderungen definieren & Mitarbeiter für sich gewinnen • Entwerfen von strategischen Projektstrukturplänen • Analyse von Projektumwelt und –risiken • Umgehen von Fallstricken bei verteilten Teams • Die fünf wichtigsten Führungstechniken • Projekt- und Fortschrittscontrolling – immer alles im Griff • Sieben Erfolgsstrategien für höhere Motivation Methoden: Fortlaufende Bearbeitung einer Fallstudie in Kleingruppen, Abschlusspräsentation der jeweiligen Fallstudie durch die Kursteilnehmer, ausführliches Feedback durch Kursteilnehmer und -leiter Dozentin: Sabine Schumann, Trainerin und Projektmanagement (GPM)

... (weiter siehe Digicampus)

Kurs Präsentationstraining (Kurs)

Veranstaltung wird online/digital abgehalten.

Wirkungsvoll präsentieren: Menschen begeistern – Emotionen erzeugen – Erfolg steigern Dieser Online-Kurs vermittelt Ihnen das nötige Know-how und gibt viele praxiserprobte Werkzeuge und Tipps an die Hand, um souverän, wirkungsvoll und begeisternd zu moderieren und zielführend zu präsentieren. Die Inhalte des praxisorientierten Präsenz-Online Trainings sind sofort in Ihrem Alltag und Studium erfolgreich einsetzbar! Inhalte: • Ablauf einer Präsentation effektiv vorbereiten und gestalten • Die 10 Module einer erfolgreichen Präsentation • Professionelle Vorstellung und Begrüßung • Souverän Auftreten und Begeisterung ausstrahlen • Körpersprache, Sprache und Stimme wirkungsvoll einsetzen • Die Zuhörer für sich gewinnen, motivieren und Interesse wecken • Tipps und Tricks zur Steigerung der Publikumsaufmerksamkeit Methodik: Aktives praxisorientiertes Online Training mit Partnerübungen im virtuellen Klassenraum und Einzelübungen. Leistungsnachweis: Anwesenheit an allen synchronen Terminen, Bear

... (weiter siehe Digicampus)

Kurs Rhetorik (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Den Zuhörer in den Bann ziehen – in Bildern sprechen. Lernen Sie die Kunst des Sprechens sowie Gedächtnisstützen, damit Sie überzeugend und frei vortragen können. Dieses besondere Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Rede. In Zukunft wird Ihre Stimme süchtig machen. Überzeugen Sie ab heute jeden durch unschlagbare Argumentationsketten. Lerninhalte: • Körpersprache – der perfekte Auftritt • Von der Idee zum fesselnden Vortrag – die optimale Herangehensweise & Umsetzung • Packende Argumentationsstrategien – Überrede nicht, überzeuge! • Wirksprache – wirken, nachwirken, im Gedächtnis bleiben. • Sprachkompetenz und Emotionale Intelligenz – damit du einen rundherum positiven Eindruck hinterlässt • Merk- & Präsentationstechniken – dein Leitfaden, um mit wenig alles vorzutragen • Deine Stimme, dein Instrument – Trainingstipps für den richtigen Ton • Sprachkompetenz – Worte, meine Fallschirme... wer euch richtig öffnet, schwebt! • Rhetorik im Alltag – B

... (weiter siehe Digicampus)

Kurs Rhetorik (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Den Zuhörer in den Bann ziehen – in Bildern sprechen. Lernen Sie die Kunst des Sprechens sowie Gedächtnisstützen, damit Sie überzeugend und frei vortragen können. Dieses besondere Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Rede. In Zukunft wird Ihre Stimme süchtig machen. Überzeugen Sie ab heute jeden durch unschlagbare Argumentationsketten. Lerninhalte: • Körpersprache – der perfekte Auftritt • Von der Idee zum fesselnden Vortrag – die optimale Herangehensweise & Umsetzung • Packende Argumentationsstrategien – Überrede nicht, überzeuge! • Wirksprache – wirken, nachwirken, im Gedächtnis bleiben. • Sprachkompetenz und Emotionale Intelligenz – damit du einen rundherum positiven Eindruck hinterlässt • Merk- & Präsentationstechniken – dein Leitfaden, um mit wenig alles vorzutragen • Deine Stimme, dein Instrument – Trainingstipps für den richtigen Ton • Sprachkompetenz – Worte, meine Fallschirme... wer euch richtig öffnet, schwebt! • Rhetorik im Alltag – B

... (weiter siehe Digicampus)

Kurs Rhetorik (Option 3) (Kurs)

Veranstaltung wird online/digital abgehalten.

Den Zuhörer in den Bann ziehen – in Bildern sprechen. Lernen Sie die Kunst des Sprechens sowie Gedächtnisstützen, damit Sie überzeugend und frei vortragen können. Dieses besondere Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Rede. In Zukunft wird Ihre Stimme süchtig machen. Überzeugen Sie ab heute jeden durch unschlagbare Argumentationsketten. Lerninhalte: • Körpersprache – der perfekte Auftritt • Von der Idee zum fesselnden Vortrag – die optimale Herangehensweise & Umsetzung • Packende Argumentationsstrategien – Überrede nicht, überzeuge! • Wirksprache – wirken, nachwirken, im Gedächtnis bleiben. • Sprachkompetenz und Emotionale Intelligenz – damit du einen rundherum positiven Eindruck hinterlässt • Merk- & Präsentationstechniken – dein Leitfaden, um mit wenig alles vorzutragen • Deine Stimme, dein Instrument – Trainingstipps für den richtigen Ton • Sprachkompetenz – Worte, meine Fallschirme... wer euch richtig öffnet, schwebt! • Rhetorik im Alltag – B

... (weiter siehe Digicampus)

Kurs Selbstmanagement & Produktivitätssysteme (Kurs)

Veranstaltung wird online/digital abgehalten.

Das Studium überdurchschnittlich gut abschließen und trotzdem noch genügend Freizeit haben? In diesem Power-Workshop werden Ihnen Instrumente zur Produktivitäts- und Effektivitätssteigerung an die Hand gegebene und vermittelt, wie Sie sich am besten selbst organisieren, um konstant Fortschritt in Ihren Projekten zu erzielen. Es wird ein im Businesskontext erprobtes Produktivitätssystem vorgestellt, das jeder Teilnehmer individuell auf seine Bedürfnisse zuschneiden kann. Am ersten (halben) Tag des Webinars werden aktuelle Methoden des Selbstmanagements und der Produktivitätsforschung kompakt vorgestellt und miteinander verknüpft. Unter Anleitung des Dozenten definieren Sie Ihre Ziele und leiten umsetzbare Projekte ab. Am zweiten (halben) Tag werden Sie ein digitales Produktivitätssystem aufbauen, das Sie bei der Erreichung der gesetzten Ziele unterstützen wird. Ein digitales Produktivitätssystem ist ein Zusammenschluss von verschiedenen Programmen, das den Anwender dabei unterstützen so

... (weiter siehe Digicampus)

Kurs Strategische Gesprächsführung (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

• Sie wollen die nächste Verhandlung für sich entscheiden? • Lernen Sie konstruktive Verhandlungstaktiken und -strategien kennen. • Erfahren Sie was, es bedeutet überzeugend zu agieren und gelungene Verhandlungen zu führen. Wir verhandeln zu jeder Zeit, nur ist es uns oftmals nicht bewusst. Dieses außergewöhnliche Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Verhandlung. Sie lernen konstruktive Verhandlungstaktiken und -strategien kennen und erfahren, wie Sie durch den Einsatz von gezielten Verhandlungstechniken wesentliche Vorteile für sich nutzen können. Sie werden zukünftig den Verhandlungspartner besser einschätzen, seine Verhandlungsstrategien erkennen und Ihre eigene Position in Verhandlungen überzeugender und nutzenorientierter darstellen können. Lerninhalte: • Psychologische

Grundlagen effektiv nutzen • Sympathie im Gespräch erzeugen • Die Basics aus der Kunst der Diplomatie & die goldenen Regeln der Gesprächsführung • Den Mittelpunkt geschic
... (weiter siehe Digicampus)

Kurs Strategische Gesprächsführung (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Oft setzen sich nicht die besten Ideen durch, sondern die, die mit dem größten Nachdruck vertreten werden. Blieben Ihre Ideen auf der Strecke? Stecken Sie in Diskussionen oft zurück, um andere nicht vor den Kopf zu stoßen und ärgern sich danach darüber? In diesem Seminar trainieren Sie das, was Ihnen wichtig ist, auch gegen Widerstände vertreten und wenn möglich durchsetzen. Sie lernen selbstbewusst für Ihre Ziele und Interessen einzustehen, ohne die Beziehung zu Ihren Gesprächspartnern auf eine Zerreißprobe zu stellen. In Rollenspielen, die an den Berufsalltag angelehnt sind, stärken Sie Ihre Persönlichkeit, indem Sie üben erfolgreich zu diskutieren, zu debattieren und auch mal selbstbewusst „Nein“ zu sagen. Das Training vermittelt Ansätze und Techniken, um in Verhandlungen nachhaltig besser und erfolgreicher abzuschneiden. Sie lernen, auch mit schwierigen Verhandlungspartnern und -Situationen umzugehen. Lerninhalte: • Bedeutung der inneren Einstellung für Selbstbewusstsein, Selbstsic
... (weiter siehe Digicampus)

Kurs Strategische Gesprächsführung (Option 3) (Kurs)

Veranstaltung wird online/digital abgehalten.

• Sie wollen die nächste Verhandlung für sich entscheiden? • Lernen Sie konstruktive Verhandlungstaktiken und -strategien kennen. • Erfahren Sie was, es bedeutet überzeugend zu agieren und gelungene Verhandlungen zu führen. Wir verhandeln zu jeder Zeit, nur ist es uns oftmals nicht bewusst. Dieses außergewöhnliche Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Verhandlung. Sie lernen konstruktive Verhandlungstaktiken und -strategien kennen und erfahren, wie Sie durch den Einsatz von gezielten Verhandlungstechniken wesentliche Vorteile für sich nutzen können. Sie werden zukünftig den Verhandlungspartner besser einschätzen, seine Verhandlungsstrategien erkennen und Ihre eigene Position in Verhandlungen überzeugender und nutzenorientierter darstellen können. Lerninhalte: • Psychologische Grundlagen effektiv nutzen • Sympathie im Gespräch erzeugen • Die Basics aus der Kunst der Diplomatie & die goldenen Regeln der Gesprächsführung • Den Mittelpunkt geschic
... (weiter siehe Digicampus)

Kurs Teamleitung (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Ziel des Seminars ist es, die Herausforderungen und Potentiale von Teams zu verstehen und nutzen zu lernen. Dafür werden Sie verschiedene Methoden kennenlernen, wie sie Ihr Team für die gemeinsamen Ziele begeistern und dorthin führen können. In interaktiven Übungen werden Sie das neugelernte Wissen vertiefen und Sie erleben direkt den Zusammenhalt und das Führungsverhalten von anderen und sich selbst. Inhalte: • Rhetorik – Ihre Gruppe für Ihre Ideen begeistern • Methoden der Moderation – Die besten Tricks, wie sie eine Gruppe moderieren und dynamische Arbeitsprozesse entstehen lassen. • Führungsstile – Entdecken Sie Ihren persönlichen Führungsstil • Konflikt- & Stressmanagement – Konflikte innerhalb des Teams vermeiden und gemeinsam entspannt ans Ziel • Zielsetzung – Wie Sie Ziele in einem Gruppenprojekt definieren Methoden: Theorie-Input, Tipps aus der Praxis, Fallbeispiele und Übungen, Reflexion, ausführliches Feedback, Diskussion
Dozenten: Max Daufrathhofer, Dr. Philipp Rodrian - be
... (weiter siehe Digicampus)

Kurs Teamleitung (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Ziel des Seminars ist es, die Herausforderungen und Potentiale von Teams zu verstehen und nutzen zu lernen. Dafür werden Sie verschiedene Methoden kennenlernen, wie sie Ihr Team für die gemeinsamen Ziele begeistern und dorthin führen können. In interaktiven Übungen werden Sie das neugelernte Wissen vertiefen und Sie erleben direkt den Zusammenhalt und das Führungsverhalten von anderen und sich selbst. Inhalte: • Rhetorik – Ihre Gruppe für Ihre Ideen begeistern • Methoden der Moderation – Die besten Tricks, wie sie eine Gruppe moderieren und dynamische Arbeitsprozesse entstehen lassen. • Führungsstile – Entdecken Sie Ihren persönlichen Führungsstil • Konflikt- & Stressmanagement – Konflikte innerhalb des Teams vermeiden und

gemeinsam entspannt ans Ziel • Zielsetzung – Wie Sie Ziele in einem Gruppenprojekt definieren Methoden: Theorie-Input, Tipps aus der Praxis, Fallbeispiele und Übungen, Reflexion, ausführliches Feedback, Diskussion
Dozent: Andreas Renner / Max Daufratshofer, Steinbeis
... (weiter siehe Digicampus)

Kurs Unternehmerisches Denken - Lean Startup (Kurs)

Veranstaltung wird online/digital abgehalten.

Fach- und Führungskräfte mit technischem, naturwissenschaftlichem, juristischem oder geistes- und sozialwissenschaftlichem Hintergrund werden in ihrem Arbeitsalltag zunehmend mit betriebswirtschaftlichen Fragen konfrontiert oder denken gar über eine Unternehmensgründung nach. In diesem Seminar lernen sie die ökonomischen Grundlagen sowie die entsprechenden Fachbegriffe kennen und können diese sofort im Rahmen einer Unternehmensgründung kompetent anwenden und praxisnah erleben. Teilnehmern mit wenig fundierten bzw. ohne betriebswirtschaftliche Vorkenntnisse bietet die Projektarbeit und das Durchspielen einer konkreten Gründung anhand der Lean Startup Methode und des Business Model Canvas einen interessanten Einstieg in ökonomische Zusammenhänge und betriebswirtschaftliche Entscheidungsparameter. Das Verständnis für unternehmerische Vorhaben sowie der sog. Unternehmergeist kann so bei Teilnehmern unterschiedlicher Zielgruppen in Teams gefördert werden. Lerninhalte: • Interaktives Gruppen
... (weiter siehe Digicampus)

Startup Challenge - Augsburg Center for Entrepreneurship (Projektstudium)

Veranstaltung wird online/digital abgehalten.

• Innovatives Entrepreneurship Format und Startup Projektstudium • Anhand einer realen unternehmerischen Entscheidungssituation können die Studierenden ihr ganzes unternehmerisches Talent testen • Die Studierenden lernen wesentliche Methoden und Konzepte zur Steuerung und Bewertung von Geschäftsmodellen, Pricing Strategien, Vertrieb und Marketing. • Experten und renommierte Startup-Mentoren begleiten die Studierenden in fachlichen Fragen. • Fachvorträge zu ausgewählten Themenstellungen runden das Seminarangebot ab; sowie die Möglichkeit mit der Augsburger Startup Szene auf Tuchfühlung zu gehen.

Startup Challenge - Augsburg Center for Entrepreneurship (Seminar) (Seminar)

Veranstaltung wird online/digital abgehalten.

• Innovatives Entrepreneurship Format und Startup Projektseminar • Anhand einer realen unternehmerischen Entscheidungssituation können die Studierenden ihr ganzes unternehmerisches Talent testen • Die Studierenden lernen wesentliche Methoden und Konzepte zur Steuerung und Bewertung von Geschäftsmodellen, Pricing Strategien, Vertrieb und Marketing. • Experten und renommierte Startup-Mentoren begleiten die Studierenden in fachlichen Fragen. • Fachvorträge zu ausgewählten Themenstellungen runden das Seminarangebot ab; sowie die Möglichkeit mit der Augsburger Startup Szene auf Tuchfühlung zu gehen.

Prüfung

Anwesenheit und aktive Übungsteilnahme im Kurs

Beteiligungsnachweis, unbenotet

Modul INF-0067: Peer-to-Peer und Cloud Computing <i>Peer-to-Peer and Cloud Computing</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen: Die erfolgreiche Teilnahme an dieser Lehrveranstaltung resultiert in fachlichen Kompetenzen in zwei Teilbereichen: Peer-to-Peer-Systeme und Cloud-Computing.</p> <p>Peer-to-Peer-Systeme Die Studierenden können verschiedene Peer-to-Peer-System-Architekturen und deren Vor- und Nachteile beschreiben. Insbesondere sind sie in der Lage, für einen gegebenen Anwendungsfall eine passende Architektur auszuwählen und diese Entscheidung zu begründen. Des Weiteren können sie gängige dezentrale Suchalgorithmen und -heuristiken einordnen und bezüglich ihrer Effizienz vergleichen. Sie verstehen abstrakte Netzwerkmodelle, und können mit deren Hilfe Peer-to-Peer-Netzwerke analysieren.</p> <p>Cloud-Computing Teilnehmer können die dem Cloud-Computing zugrunde liegenden Technologien unterscheiden und basierend auf diesem Wissen Cloud-Computing-Architekturen analysieren. Sie kennen die Wichtigkeit von Lastverteilung im Rahmen von großen Cloud-Computing-Anwendungen und können die verschiedenen Ansätze dafür erklären und gegeneinander abwägen. Die Studierenden kennen moderne Ansätze zu nebenläufigen Berechnungen über große Datenmengen und können diese auf einfache Probleme anwenden. Sie sind in der Lage, einfache Anwendungen mithilfe von Platform-as-a-Service-Frameworks zu entwickeln.</p> <p>Schlüsselqualifikationen Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Peer-to-Peer und Cloud Computing (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		

Inhalte:

Dieses Modul vermittelt Ansätze zur Organisation von komplexen abstrahierten IT-Infrastrukturen, die dynamisch an wechselnde Nutzungsbedingungen angepasst werden können und Dienste wie Rechenkapazität, Datenspeicher, Netzkapazitäten oder Software auf verschiedenen Ebenen zur Verfügung stellen. Dazu werden die Anforderungen, Eigenschaften und Ziele solcher Systeme definiert und diskutiert. Darüber hinaus werden Konzepte aus dem Bereich der Systemarchitekturen und Ansätze aus dem Bereich selbstorganisierender Algorithmen dargestellt und bewertet. In allen Teilen werden Bezüge zu konkreten Anwendungsgebieten gegeben.

Literatur:

- aktuelle wissenschaftliche Paper
- Mahlmann und Schindelhauer: Peer-to-Peer Netzwerke - Algorithmen und Methoden, Springer 2007
- Antonopoulos und Gillam: Cloud Computing - Principles, Systems and Applications, Springer 2010

Modulteil: Peer-to-Peer und Cloud Computing (Übung)

Lehrformen: Übung

Sprache: Deutsch

Angebotshäufigkeit: unregelmäßig (i. d. R. im WS)

SWS: 2

Prüfung

Peer-to-Peer und Cloud Computing (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Beschreibung:

Die Prüfung kann jedes Semester zu Beginn und Ende der vorlesungsfreien Zeit abgelegt werden.

Modul INF-0069: Weiterführende Betriebssystemkonzepte <i>Advanced Operating System Concepts</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen: Die Veranstaltung vermittelt den Studierenden bei erfolgreicher Teilnahme fundierte fachliche Kenntnisse über moderne Betriebssysteme und die dahinterstehenden grundlegenden Konzepte. Sie können danach die Geschichte der Betriebssysteme wiedergeben und deren primäre Typen erläutern. Der theoretische Teil konzentriert sich auf die Vertiefung des Verständnisses von Mechanismen aus den Bereichen Multiprogramming, Interprozesskommunikation, Speicher-Management und Virtualisierung. Die Studierenden können die kennengelernten Ansätze hinsichtlich ihrer Eignung für einen bestimmten Einsatzzweck beurteilen. Der praktische Teil befasst sich mit dem Transfer und der Anwendung des im theoretischen Teil vermittelten Wissens. Sie benutzen Programme und Programmierschnittstellen aus dem Unix-Umfeld, um ein erarbeitetes Lösungskonzept mit dem Schwerpunkt Ein-/Ausgabe durch einen Programmcode zu realisieren.</p> <p>Schlüsselqualifikationen: Qualitätsbewusstsein, Akribie; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Kenntnisse der Vor-/Nachteile von Entwurfsalternativen, Bewertung im jeweiligen Anwendungszusammenhang; Fachspezifische Vertiefungen; Umsetzen fachlicher Lösungskonzepte in Programme und Modelle.</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vorlesung (Präsenzstudium) 60 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Weiterführende Betriebssystemkonzepte (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		

Inhalte:

Die Vorlesung beinhaltet u. a. folgende Themen:

- Grundlagen eines Betriebssystems
 - Aufgaben
 - Struktur und Aufbau
 - Historie
 - Shell
- Prozesse, Threads und Interrupts
 - Prozesse und Threads
 - HW-Interrupts
- Scheduling
 - Grundlagen des Scheduling
 - Echtzeit-Scheduling
- Speicher
 - Interaktion
 - Speicherkonzepte
 - Dateisysteme
- Virtualisierung

Literatur:

- Folien
- Andrew S. Tanenbaum: "Moderne Betriebssysteme", Pearson Studium, ISBN:978-3-8273-7342-7

Modulteil: Weiterführende Betriebssystemkonzepte (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Prüfung

Weiterführende Betriebssystemkonzepte

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Beschreibung:

Die Prüfung kann jedes Semester zu Beginn und Ende der vorlesungsfreien Zeit abgelegt werden.

Modul INF-0147: Prozessorarchitektur <i>Processor Architecture</i>		5 ECTS/LP
Version 2.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer Dr. Martin Frieb		
<p>Lernziele/Kompetenzen: Die Studierenden verstehen die Prinzipien des Aufbaus folgender Prozessoren auf einem fortgeschrittenen, praxisorientierten, aber wissenschaftlichen Niveau: superskalare Mikroprozessoren, Multi- und Manycore-Prozessoren, Signalprozessoren, Smartphone-Prozessoren und GPUs. Außerdem lernen sie aktuelle Konzepte der Prozessorarchitektur kennen und können Vor-/Nachteile aktueller und zukünftiger Prozessoren anhand ihres internen Aufbaus einschätzen und beurteilen. Sie verstehen die Funktionsweise von verschiedenen Komponenten von Mikroprozessoren und deren Zusammenspiel. Somit ist es den Studierenden möglich, den Einfluss verschiedener Architekturerweiterungen auf das Gesamtsystem einzuordnen. Ebenso können sie unterscheiden, wann es sinnvoll ist, speichergekoppelte, nachrichtengekoppelte oder datenparallele Prozessoren einzusetzen, und differenzieren zwischen Techniken für energieeffiziente oder hochperformante Prozessoren. In der Übung simulieren die Studierenden verschiedene Prozessorarchitekturen und Sprungvorhersagetechniken und bewerten deren Einfluss auf die Laufzeit und den Energieverbrauch.</p> <p>Schlüsselqualifikationen: Analytisch-methodische Kompetenz, Abwägung von Lösungsansätzen, Präsentation von Lösungen von Übungsaufgaben; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein, Akribie; Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen; Fertigkeit, mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen anzustellen; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Modul Systemnahe Informatik (INF-0138) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Prozessorarchitektur (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Vorlesung "Prozessorarchitektur" vertieft die Techniken superskalärer Mikroprozessoren und aktueller Multicore-Prozessoren. Dabei werden die Pipelinestufen detailliert behandelt, mehrfädige Prozessoren und Multicores gegenübergestellt sowie aktuelle Beispielprozessoren vorgestellt. Außerdem wird aus der Forschung an Manycores und Echtzeit-Multicores berichtet.</p>		

Literatur:

- Uwe Brinkschulte, Theo Ungerer, Mikrocontroller und Mikroprozessoren, Springer Verlag, Heidelberg, dritte Auflage 2010
- John L. Hennessy, David A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 5. Auflage, 2011

Modulteil: Prozessorarchitektur (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Prozessorarchitektur

Klausur / Prüfungsdauer: 60 Minuten

Modul INF-0149: Praktikum Eingebettete Systeme <i>Practical Module Embedded Systems</i>		5 ECTS/LP
Version 2.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen: Die Studierenden erwerben Kompetenzen in den folgenden Bereichen auf einem fortgeschrittenen, praxisorientierten, aber wissenschaftlichen Niveau: selbstständiges Arbeiten mit Mikrocontrollern, Datenblättern und Spezifikationen, Anbindung von analoger und digitaler Peripherie, Entwurf und Modellierung von eingebetteter Software mit Zustandsdiagrammen sowie deren Transformation in Code. Weitere Schwerpunkte sind die Konfiguration von sequentiellen Schnittstellen sowie Scheduling und task-basierte Programmierung.</p> <p>Durch die Implementierung von Aufgaben für einen Mikrocontroller setzen sie die erlernten Konzepte direkt in die Praxis um. Dabei steht Interaktion mit Sensoren und Aktoren sowie die Kommunikation mit anderen Systemteilen im Vordergrund. Dazu können sie unterschiedliche Arten der Ablaufsteuerung unterscheiden und anwenden. In einer abschließenden Projektphase lernen die Studierenden, eine komplexe Aufgabenstellung im Team zu planen, nach einem selbst entwickelten fundierten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zur verständlichen Darstellung und Dokumentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein, Akribie; Projektgebundene Arbeit und Zeitmanagement; Auswahl und sichere Anwendung geeigneter Methoden; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Selbstreflexion</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Praktikum (Präsenzstudium)</p>		
Voraussetzungen: Kenntnisse in C.		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Eingebettete Systeme		
Lehrformen: Praktikum		
Sprache: Deutsch		
SWS: 4		
<p>Inhalte: In dem Praktikum "Eingebettete Systeme" sollen die Einschränkungen und Herausforderungen für das Programmieren von eingebetteten Systemen erlernt werden. Als Plattform dient ein Staubsaugerroboter (ROOMBA) und ein daran angeschlossener Mikrocontroller zur Steuerung des ROOMBA. Die Programmierung erfolgt hardwarenah in C und die erstellten Programme sollen die Sensoren des ROOMBA auslesen und entsprechende Aktuatoren stellen. Dabei sollen insbesondere die Herausforderungen eingebetteter Systeme, wie z.B. das Zeitverhalten der Software sowie das Arbeiten mit Datenblättern kennengelernt werden.</p> <p>In einer Projektphase sollen die anfänglich erlernten Grundkenntnisse vertieft werden und komplexere Steuerungsprogramme entwickelt werden, z.B. ein autonomer Explorator oder ein "ROOMBA-Rennen" durch ein Labyrinth. Die Projekte werden einzeln oder im Team bearbeitet, dokumentiert und am Ende des Praktikums präsentiert.</p>		

Literatur:

- Marwedel, Wehmeyer: Eingebettete Systeme, Springer Verlag, Heidelberg, 2007
- Wiegmann: Softwareentwicklung in C für Mikroprozessoren und Mikrocontroller, 6. Auflage, VDE Verlag, Berlin, 2011

Prüfung

Praktikum Eingebettete Systeme

praktische Prüfung

Modul INF-0150: Hardware-Entwurf <i>Hardware Design</i>		8 ECTS/LP
Version 2.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer Dr. Martin Frieb		
<p>Lernziele/Kompetenzen: Die Studierenden erwerben Kompetenzen in den folgenden Bereichen auf einem fortgeschrittenen, praxisorientierten, aber wissenschaftlichen Niveau: Entwurfsprozess für digitale Schaltungen, Schaltungslogik und Gatter, physikalische Grundlagen der elektronischen Bauteile, Beschreibung von Hardware mit einer Hardwarebeschreibungssprache. Zunächst lernen die Studierenden die Verknüpfung logischer Gatter und bauen einen Halb- und einen Volladdierer. Sie verstehen den Entwicklungsprozess für digitale Schaltungen und wenden diesen direkt praktisch an, indem sie ihren eigenen RISC-V-Prozessor entwickeln. Sie modellieren und implementieren ihn selbstständig mit Hilfe der Hardwarebeschreibungssprache VHDL. Dazu lernen sie die Vor- und Nachteile von schematischer und textueller Hardware-Beschreibung kennen und können entscheiden, wann es sinnvoll ist, welche Variante einzusetzen. Darüber hinaus kombinieren sie synchrone und asynchrone Prozesse, um ein gutes Zusammenspiel der Komponenten ihres selbstgebauten Mikroprozessors zu erreichen. Schließlich beurteilen die Studierenden die Effizienz ihrer Implementierung anhand der erzielten Taktfrequenz und des benötigten Hardware-Aufwandes. In einer abschließenden Projektphase lernen sie, eine komplexe Aufgabenstellung zu planen, nach einem selbst entwickelten fundierten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zur verständlichen Darstellung und Dokumentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein, Akribie; Projektgebundene Arbeit und Zeitmanagement; Auswahl und sichere Anwendung geeigneter Methoden; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Selbstreflexion</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen: Modul Systemnahe Informatik (INF-0138) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Hardware-Entwurf (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Veranstaltung "Hardware-Entwurf" stellt Methoden des logischen Entwurfs digitaler Schaltungen dar, angefangen von der abstrakten Beschreibung in einer Hardwarebeschreibungssprache (wie VHDL) bis hin zur physikalischen Implementierung auf Transistorebene. Im praktischen Teil der Veranstaltung wird die Hardwareentwicklung am Beispiel einer fünfstufigen Prozessorpipeline veranschaulicht. Das Ergebnis ist ein in VHDL entwickelter lauffähiger Prozessor für ein FPGA-Prototypenboard</p>		

Literatur:

- Uwe Brinkschulte, Theo Ungerer, Mikrocontroller und Mikroprozessoren, Springer Verlag, Heidelberg, dritte Auflage 2010
- John L. Hennessy, David A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 5. Auflage, 2011

Modulteil: Hardware-Entwurf (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Prüfung

Hardware-Entwurf

praktische Prüfung

Beschreibung:

Erfolgreiche Praktikumsteilnahme, Projektvorstellung am Ende des Semesters

Modul INF-0216: Vertiefte Multicore-Programmierung <i>Advanced Multicore Programming</i>		8 ECTS/LP
Version 2.0.0 (seit SoSe20 bis WS20/21) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen: Die Studierenden erwerben Kompetenzen auf einem fortgeschrittenen, wissenschaftlichen Niveau in folgenden Bereichen: Parallelprogrammierung von speichergekoppelten Systemen in C++11 und OpenMP, Entwurf von lock-freien Algorithmen, Programmierung eines Hardware-Transaktionsspeicher-Systems am Beispiel von Intel TSX, nachrichtengekoppelte Programmierung mit dem Message-Passing-Interface (MPI), Programmierung von GPU-Beschleunigerkarten mit Nvidia CUDA.</p> <p>Sie können die Vor- und Nachteile der unterschiedlichen Programmiermodelle und Architekturen einschätzen, und parallele Programme in den jeweiligen Sprachen analysieren. Durch die praktischen Übungen setzen die Studierenden die erlernten Konzepte direkt in die Praxis um. Dabei erwerben die Studierenden fortgeschrittene Programmierkenntnisse in C++11, Intel TSX, MPI, OpenMP und CUDA. Sie können selbständig das Laufzeitverhalten der jeweiligen Programmiermodelle bewerten und optimieren. In einer abschließenden Projektphase lernen die Studierenden, eine komplexe Aufgabenstellung im Team zu planen, nach einem selbst entwickelten Lösungsansatz zu bearbeiten und die Resultate in wissenschaftlicher Weise im Plenum zu diskutieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Analytische-methodische Kompetenz im Bereich der Multicore-Programmierung von CPUs und GPUs; Projektgebundene Arbeit und Zeitmanagement; Auswahl und sichere Anwendung geeigneter Methoden; Fähigkeit, vorhandenes Wissen selbständig zu kombinieren und zu erweitern; Selbstreflexion</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 60 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Vertiefte Multicore-Programmierung (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Vorlesung vertieft verschiedene Techniken der Parallelprogrammierung für aktuelle Multicore-Prozessoren und Grafikkarten. Nach einer grundlegenden Einführung in Threads, Synchronisationskonstrukte und weiterführende Konzepte der Parallelprogrammierung in C++11 werden weitere parallele Programmiermodelle behandelt.</p>		

Literatur:

- M. Herlihy, N. Shavit: The Art of Multiprocessor-Programming, Morgan Kaufmann, 2012, ISBN: 978-0123973375
- M. McCool, J. Reinders, A. D. Robison: Structured Parallel Programming: Patterns for Efficient Computation, Morgan Kaufmann, 2012, ISBN: 978-0124159938
- T. Rauber, G. Runger: Parallele Programmierung, Springer Verlag, 2007, ISBN: 978-3540465492
- es werden die jeweils neuesten APIs/Unterlagen aus dem Internet verwendet

Modulteil: Vertiefte Multicore-Programmierung (ubung)

Lehrformen: ubung

Sprache: Deutsch

SWS: 4

Inhalte:

Ziel der ubung ist es, den Umgang mit den unterschiedlichen Programmiermodellen sowie Performanzanalyse- und Debugging-Techniken in praktischen Beispielen zu vertiefen. Die ubung wird durch eine Projektphase abgeschlossen, die es den Studierenden ermoglicht, die behandelten Programmier Techniken in einem umfangreicheren Projekt selbstandig anzuwenden, Ergebnisse auszuwerten und zu prasentieren.

Prufung

Vertiefte Multicore-Programmierung

Portfolioprufung

Beschreibung:

Projektvorstellung und -abnahme, Fragen zu Vorlesung und ubung

Modul INF-0297: Praktikum Prozessorbau <i>Practical Module Processor Design</i>		5 ECTS/LP
Version 2.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer Dr. Martin Frieb		
Lernziele/Kompetenzen: Die Studierenden erwerben Kompetenzen in den folgenden Bereichen auf einem fortgeschrittenen, praxisorientierten, aber wissenschaftlichen Niveau: Entwurfsprozess für digitale Schaltungen, Schaltungslogik und Gatter, physikalische Grundlagen der elektronischen Bauteile, Beschreibung von Hardware mit einer Hardwarebeschreibungssprache. Zunächst lernen die Studierenden die Verknüpfung logischer Gatter und bauen einen Halb- und einen Volladdierer. Sie verstehen den Entwicklungsprozess für digitale Schaltungen und wenden diesen direkt praktisch an, indem sie ihren eigenen RISC-V-Prozessor entwickeln. Sie modellieren und implementieren ihn selbstständig mit Hilfe der Hardwarebeschreibungssprache VHDL. Dazu lernen sie die Vor- und Nachteile von schematischer und textueller Hardware-Beschreibung kennen und können entscheiden, wann es sinnvoll ist, welche Variante einzusetzen. Darüber hinaus kombinieren sie synchrone und asynchrone Prozesse, um ein gutes Zusammenspiel der Komponenten ihres selbstgebauten Mikroprozessors zu erreichen. Schließlich beurteilen die Studierenden die Effizienz ihrer Implementierung anhand der erzielten Taktfrequenz und des benötigten Hardware-Aufwandes. In einer abschließenden Projektphase lernen sie, eine komplexe Aufgabenstellung zu planen, nach einem selbst entwickelten fundierten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren. Schlüsselqualifikationen: Fertigkeit zur verständlichen Darstellung und Dokumentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein, Akribie; Projektgebundene Arbeit und Zeitmanagement; Auswahl und sichere Anwendung geeigneter Methoden; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Selbstreflexion		
Arbeitsaufwand: Gesamt: 150 Std. 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Praktikum (Präsenzstudium)		
Voraussetzungen: Modul Systemnahe Informatik (INF-0138) - empfohlen Modul Prozessorarchitektur (INF-0147) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Praktikum Prozessorbau Lehrformen: Praktikum Sprache: Deutsch SWS: 4		
Inhalte: Die Veranstaltung "Prozessorbau" stellt Methoden des logischen Entwurfs digitaler Schaltungen dar, angefangen von der abstrakten Beschreibung in einer Hardwarebeschreibungssprache (wie VHDL) bis hin zur physikalischen Implementierung auf Transistorebene. Im praktischen Teil der Veranstaltung wird die Hardwareentwicklung am Beispiel einer fünfstufigen Prozessorpipeline veranschaulicht. Das Ergebnis ist ein in VHDL entwickelter lauffähiger Prozessor für ein FPGA-Prototypenboard		

Literatur:

- Uwe Brinkschulte, Theo Ungerer, Mikrocontroller und Mikroprozessoren, Springer Verlag, Heidelberg, dritte Auflage 2010
- John L. Hennessy, David A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 5. Auflage, 2011

Prüfung

Praktikum Prozessorbau

praktische Prüfung

Modul INF-0309: Echtzeitsysteme <i>Real-Time Systems</i>		8 ECTS/LP
Version 1.5.0 (seit WS19/20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen:</p> <p>Die Vorlesung vermittelt grundlegende und weiterführende Kenntnisse zu Echtzeitsystemen wie sie in nahezu allen eingebetteten Systemen, jedoch speziell in den Bereichen Automobil, Luft-/Raumfahrt und Robotik vorkommen. Die theoretischen Grundlagen werden sich am aktuellen Forschungsstand orientieren und eine weitere Beschäftigung der Studierenden mit dem Thema eingebettete Echtzeitsysteme auf wissenschaftlichen Niveau ermöglichen.</p> <p>Die Vorlesung wird den Studierenden die Fähigkeit vermitteln verschiedene eingebettete Systeme anhand ihrer Echtzeitanforderungen zu unterscheiden und einzuordnen. Die Studierenden lernen aktuelle Methoden zur Validierung des Zeitverhaltens anzuwenden, zu vergleichen, und im Hinblick auf eine mögliche Zertifizierung des Zeitverhaltens kritisch zu analysieren. Dies beinhaltet die Optimierung und Auswahl von Echtzeit-Schedules und deren Verifikation. Die Vorlesung behandelt zudem verschiedene Prozessortypen, und wird näher auf die Besonderheiten von Einkern und Mehrkern-Prozessoren im Echtzeitbereich eingehen. Die Studierenden werden hierbei in die Lage versetzt, Prozessoren anhand ihrer Eignung für Echtzeitsysteme zu klassifizieren und den Einfluss von Entwurfsentscheidungen auf das Echtzeitverhalten und die Analyse des Echtzeitverhaltens zu untersuchen. Der Lehrstoff wird anhand von Fallstudien aus den Bereichen Automobil und Luftfahrt exemplifiziert und von den Studierenden anhand eines einfachen Echtzeitsystems angewendet.</p> <p>Schlüsselqualifikationen:</p> <p>Analytisch-methodische Kompetenz, Abwägung von Lösungsansätzen, Präsentation von Lösungen von Übungsaufgaben; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein, Akribie; Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 240 Std.</p> <p>30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>60 Std. Vorlesung (Präsenzstudium)</p> <p>30 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen:</p> <p>keine</p>		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Echtzeitsysteme (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Deutsch / Englisch</p> <p>SWS: 4</p>		

Inhalte:

- WCET Analyse
- Scheduling Analyse
- Programmierung von Echtzeitsystemen
- Prozessoren für Echtzeitsysteme
- Echtzeitbetriebssysteme
- Zertifizierung von Echtzeitsystemen

Literatur:

- Sanjoy Baruah, Marko Bertogna, Giorgio Buttazzo, Multiprocessor Scheduling for Real-Time Systems, Springer, 2015.
- Giorgio Buttazzo, Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications, Springer, 2011.
- Heinz Wörn, Uwe Brinkschulte, Echtzeitsysteme, Springer Verlag, Berlin/Heidelberg, 2005
- Uwe Brinkschulte, Theo Ungerer, Mikrocontroller und Mikroprozessoren, Springer Verlag, Heidelberg, dritte Auflage 2010

Zugeordnete Lehrveranstaltungen:

Echtzeitsysteme (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Echtzeitsysteme sind Computer Systeme, deren Korrektheit nicht nur über das korrekte Ergebnis einer Berechnung, sondern auch über das korrekte Zeitverhalten definiert wird. Vereinfacht gesagt: Wenn ein eingebettetes System zu spät reagiert, ist es genauso unbrauchbar, wie wenn es gar nicht oder falsch reagiert hätte. Ein Airbag zum Beispiel sollte aufgehen, bevor der Kopf des Fahrers auf das Lenkrad knallt. Ein selbstfahrendes Fahrzeug muss ebenso rechtzeitig autonom bremsen, bevor ein Unfall geschehen ist, und eine Kaffeemaschine sollte aufhören Kaffee auszuschenken, bevor die Tasse überläuft. In dieser Vorlesung beschäftigen wir uns mit dem Zeitverhalten eben solcher eingebetteter Echtzeitsysteme. Wir lernen wie wir das Zeitverhalten dieser Systeme beeinflussen und validieren können und wir lernen, welchen Einfluss verschiedene Entwurfsentscheidungen auf das Zeitverhalten haben.

... (weiter siehe Digicampus)

Modulteil: Echtzeitsysteme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Echtzeitsysteme (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Echtzeitsysteme

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0319: Praktikum Interdisziplinäres Projekt Ingenieurinformatik <i>Interdisciplinary Project Engineering Informatics</i>		6 ECTS/LP
Version 1.0.0 (seit WS19/20) Modulverantwortliche/r: Prof. Dr.-Ing. Johannes Schilp Prof. Dr.-Ing. Lars Mikelsons, Prof. Dr.-Ing. Christoph Ament		
<p>Lernziele/Kompetenzen: Die Studierenden beschäftigen sich mit einer fächerübergreifenden Aufgabenstellung aus dem Bereich der Produktionsinformatik, Regelungstechnik und Mechatronik. Sie setzen komplexe Verfahren und Techniken ein, die teilweise in den einzelnen Vorlesungen bereits theoretisch behandelt wurden. Das praxisnahe Projekt orientiert sich an einer Studenten-Challenge, wie beispielweise der Sioux Mechatronics Trophy oder dem James Dyson Award, und wird in Kleingruppen bearbeitet. Der Anwendungsfall erfordert die Bewertung und Übertragung der Konzepte und Methoden sowie deren interdisziplinäre Kombination.</p> <p>Der zeitliche Ablauf dieses Praktikums wird in Digicampus bekannt gegeben, ebenso wie die spezifische Aufgabenstellung der Challenge einschließlich der Voraussetzungen und der Lehrstuhleteiligungen.</p> <p>Schlüsselqualifikationen: Team- und Kommunikationsfähigkeit, strukturiertes und gewissenhaftes Arbeiten, anwendungsorientierte Problemlösung, Ergebnisbewertung und -dokumentation, Abwägen von Lösungsansätzen, Fertigkeit zum logischen, analytischen und konzeptionellen Denken, Abstraktionsfähigkeit</p>		
<p>Arbeitsaufwand: Gesamt: 180 Std. 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Praktikum (Präsenzstudium)</p>		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Modulteil: Praktikum Interdisziplinäres Projekt Ingenieurinformatik Lehrformen: Praktikum Sprache: Deutsch SWS: 4 ECTS/LP: 6.0</p>		
Inhalte: Die Studenten bearbeiten in Kleingruppen anwendungsorientierte Aufgaben zu Themenbereichen im industriellen Umfeld.		
Literatur: wird im jeweiligen Semester bekannt gegeben.		
<p>Prüfung Praktikum Interdisziplinäres Projekt Ingenieurinformatik praktische Prüfung</p>		

Modul INF-0335: Safety-Critical Systems <i>Safety-Critical Systems</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen:</p> <p>Die Studierenden lernen die Fehlertoleranzanforderungen in eingebetteten Systemen kennen und ordnen diese anhand der zugrundeliegenden Vorschriften und Normen (z.B. ISO 26262) zum Entwurf sicherheitsrelevanter Systeme ein.</p> <p>Sie können die Ursachen und Effekte von Hardware-Fehlern beschreiben und sind in der Lage, unterschiedliche Redundanztechniken zu analysieren und zu bewerten.</p> <p>Weitere Schwerpunkte der Vorlesung sind die Anwendung der stochastischen Grundlagen der Fehlerrechnung, das Analysieren und Modellieren von Rechensystemen mit Zuverlässigkeitsblockdiagrammen und Fehlerbäumen, sowie das Differenzieren unterschiedlicher Redundanzarten und deren Verwendung in verschiedenen Hard- und Software-Techniken zur Fehlererkennung.</p> <p>Außerdem werden Techniken zur Fehlerkorrektur, Fehlerdiagnose und Rekonfiguration vorgestellt und untersucht. Die Studierenden lernen verschiedene Konzepte fehlertolerierender Rechensysteme aus Theorie und Praxis zu klassifizieren und zu vergleichen, wie z.B. Lockstep-Ausführung, Prozess-Redundanz, ECC-Speicher-Implementierungen oder Kontrollflussüberprüfung.</p> <p>In den praktischen Übungen wenden die Studierenden die Methoden der Modellierung und Fehlerrechnung an verschiedenen Aufgaben an, arbeiten ihre Lösungen aus und vergleichen auf qualitativer Ebene verschiedene Fehlertoleranzkonzepte aus Forschung und Praxis.</p> <p>Schlüsselqualifikationen: Analytisch-methodische Kompetenz in der Analyse fehlertolerierender Rechensysteme, Abwägung von Lösungsansätzen, Verständliche Präsentation von Lösungen von Übungsaufgaben, Fähigkeit, vorhandenes Wissen selbstständig zu erweitern, Selbstreflexion</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen:		
keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Safety-Critical Systems (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		

Inhalte:

Die Vorlesung betrachtet Methoden zur Einhaltung der funktionalen Sicherheit in Rechensystemen, sowie deren Entwurf und Analyse. Zunächst werden verschiedene Fehlerarten charakterisiert und die Bedeutung von Fehlermodellen hervorgehoben. Danach werden unterschiedliche Hardware- und Software-Methoden zur Erkennung und Tolerierung von Fehlern vorgestellt. Die diskutierten Maßnahmen beziehen sich nicht nur auf strukturelle, sondern auch auf zeitliche und informationelle Redundanz (fehlertolerierende Codes). Nach einem kurzen Repetitorium der Wahrscheinlichkeitsrechnung und Kombinatorik werden verschiedene Analysemethoden wie klassische Wahrscheinlichkeitsrechnung, Zuverlässigkeits-Blockdiagramme, Markovketten, FMEA und Fehlerbäume vorgestellt, Unterschiede hervorgehoben und anhand praktischer Beispiele erläutert. Verschiedene Methoden zur Fehlerinjektion, die zur Bewertung von fehlertolerierenden Systemen notwendig sind, werden kurz besprochen.

Literatur:

- D. Sorin: Fault Tolerant Computer Architecture, Morgan and Claypool, 2009
- S. Mukherjee: Architecture Design for Soft Errors, Morgan Kaufmann, 2008
- I. Koren, C.M. Krishna: Fault-Tolerant Systems, Morgan Kaufmann, 2007

Modulteil: Safety-Critical Systems (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Safety-Critical Systems

Klausur / Prüfungsdauer: 60 Minuten

Modul INF-0368: Embedded Hardware <i>Embedded Hardware</i>		5 ECTS/LP
Version 1.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen:</p> <p>Die Vorlesung vermittelt grundlegende und weiterführende Kenntnisse zu Hardwarekomponenten (z.B. GPIO, Hardware-Timer, ...) wie sie in nahezu allen eingebetteten Systemen vorkommen. Außerdem werden die Funktionsprinzipien von verschiedenen Sensoren und Aktuatoren vorgestellt. Ein besonderer Fokus liegt dabei auf deren Einsatz zur Interaktion mit der physikalischen Welt und auf Besonderheiten bei deren Anwendung. Die theoretischen Grundlagen werden sich am aktuellen Forschungsstand orientieren und eine weitere Beschäftigung der Studierenden mit dem Thema eingebetteter Hardware auf wissenschaftlichen Niveau ermöglichen.</p> <p>Die Vorlesung wird den Studierenden die Fähigkeit vermitteln verschiedene eingebettete Systeme anhand ihrer Hardwareanforderungen zu unterscheiden und einzuordnen. Die Studierenden lernen aktuelle Methoden zur Programmierung der Komponenten anzuwenden, zu vergleichen, und im Hinblick auf eine effiziente Nutzung kritisch zu analysieren. Dies beinhaltet die Auswahl und Beurteilung von für das System relevanten Hardwarekomponenten, sowie die Entwicklung einer für ein bestehendes Problem optimierten Konfiguration der Komponenten. Die Vorlesung behandelt zudem verschiedene Sensoren und Aktuatoren, und wird näher auf die Umwandlung zwischen einer analogen, physikalischen Größe und einer digitalen, elektrischen Größe eingehen. Die Studierenden werden hierbei in die Lage versetzt, physikalische Größen anhand ihrer Eigenschaften zu klassifizieren und eine Entscheidung über die für die Umwandlung benötigten Sensoren, Aktuatoren und Hardwarekomponenten zu treffen.</p> <p>Der Lehrstoff wird anhand von Fallstudien z.B. aus den Bereich Automobil exemplifiziert und von den Studierenden anhand eines einfachen eingebetteten Systems angewendet.</p> <p>Schlüsselqualifikationen: Analytisch-methodische Kompetenz, Abwägung von Lösungsansätzen, Präsentation von Lösungen von Übungsaufgaben; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein, Akribie; Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Embedded Hardware (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Deutsch / Englisch</p> <p>SWS: 2</p>		

Inhalte:

- Modellierung von Sensoren und Aktuatoren
- Grundlagen in Elektronik
- Hardwarekomponenten von eingebetteten Prozessoren und SoC
- Serielle Schnittstellen
- Sensoren und Eingabegeräte
- Aktuatoren und Ausgabegeräte

Literatur:

- Yifeng Zhu, Embedded Systems with ARM Cortex-M Microcontrollers in Assembly Language and C, E-Man Press LLC, 2017
- Edward Ashford Lee, Sanjit Arunkumar Seshia, Introduction to Embedded Systems - A Cyber-Physical Systems Approach, MIT Press, 2017
- Elecia White, Making Embedded Systems, O'Reilly Media, 2012
- John Catsoulis, Designing Embedded Hardware, O'Reilly Media, 2005
- Rüdiger R. Asche, Embedded Controller: Grundlagen und praktische Umsetzung für industrielle Anwendungen, Springer, 2016
- Hermann Winner, Stephan Hakuli, Felix Lotz, Christina Singer Hrsg., Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort, Springer, 2015
- Bernhard Grimm, Gregor Häberle, Heinz Häberle, u.a., Fachkunde Industrieelektronik und Informationstechnik, Europa Lehrmittel, 2003

Zugeordnete Lehrveranstaltungen:

Embedded Hardware (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Eingebettete Systeme sind Computer Systeme, deren vorrangige Aufgabe es ist die umgebende physikalische und technische Welt zu erfassen bzw. zu beeinflussen. Diese kann beispielsweise durch regelmäßiges Messen physikalischer Größen oder durch die Steuerung bzw. Regelung von technischen Anlagen geschehen. Um diese Aufgaben zu bewerkstelligen benötigen eingebettete Prozessoren diverse Hardwarekomponenten, die es ihnen ermöglicht mit der Außenwelt zu interagieren. Zusätzlich werden Sensoren und Aktuatoren benötigt. Sie übernehmen die Umwandlung von physikalischen Größen in elektrische Signale und umgekehrt. In dieser Vorlesung beschäftigen wir uns sowohl mit den beschriebenen Hardwarekomponenten als auch mit verschiedenen Sensoren und Aktuatoren. Wir lernen wie ein Prozessor mit Hilfe dieser Komponenten mit seiner Umgebung interagieren kann und welchen Einfluss die Komponenten auf die Wandlung von physikalischen Größen zu einem digitalen Wert (und umgekehrt) haben kann.
 ... (weiter siehe Digicampus)

Modulteil: Embedded Hardware (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Embedded Hardware (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Embedded Hardware

Klausur / Prüfungsdauer: 60 Minuten

Modul INF-0037: Praktikum Automotive Software Engineering <i>Practical Module Automotive Software Engineering</i>		6 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
Lernziele/Kompetenzen: Nach der Teilnahme am Praktikum Automotive Software Engineering verstehen die Studierenden praxisnaher Problemstellungen höherer Komplexität im Bereich der Entwicklung und Absicherung von Fahrerassistenzsystemen mit aktuellen Methoden und Tools der modellbasierten Entwicklung eingebetteter Systeme. Die Studierenden erlangen tiefere fachspezifische als auch fächerübergreifende Kenntnisse und Fähigkeiten, beispielsweise aus der Regelungstechnik, Fahrphysik und Mathematik. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren, zu präsentieren und verständlich zu dokumentieren.		
Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Projektmanagementfähigkeiten		
Arbeitsaufwand: Gesamt: 180 Std. 90 Std. Praktikum (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)		
Voraussetzungen: Empfohlen wird die Teilnahme an einem der beiden links aufgeführten Seminare. Modul Seminar Grundlagen des Software Engineering für Automotive Systems (BA) (INF-0027) - empfohlen Modul Seminar Grundlagen des Software Engineering für Automotive Systems (MA) (INF-0040) - empfohlen		
Angebotshäufigkeit: unregelmäßig (i. d. R. im WS)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Automotive Software Engineering (MA) Lehrformen: Praktikum Sprache: Deutsch Angebotshäufigkeit: jedes Wintersemester SWS: 6		

Inhalte:

Im Automotive-Praktikum lernen die Teilnehmer wie verschiedene ausgewählte Funktionen innerhalb von Fahrzeugen simuliert und analysiert werden können. In einem zweitägigen Einführungskurs werden die benötigten theoretischen Grundlagen für die Bearbeitung der Praxisaufgabe gelegt. Während des Einführungskurses wird das Modell eines Antiblockiersystems (ABS) auf realen Steuergeräten behandelt. Die Teilnehmer lernen dabei u.a. die im Automotive-Umfeld häufig eingesetzte Modellierungswerkzeug-Kombination „Matlab/Simulink“ sowie das graphische Simulations- und Analyse-Tool „CarMaker“ kennen und erhalten einen praktischen Einblick in die Funktionsweise von FlexRay-Bussystemen.

Nach dem Einführungskurs soll in Gruppen von zwei bis drei Teilnehmern das Modell eines ACC-Systems (Adaptive Cruise Control) erstellt, simuliert und verifiziert werden.

Literatur:

abhängig vom Thema

Prüfung

Praktikum Automotive Software Engineering (MA)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0051: Algorithmen für NP-harte Probleme <i>Algorithms for NP-Hard Problems</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
Lernziele/Kompetenzen: Kenntnis verschiedener algorithmischer Lösungsansätze für NP-harte Probleme und die Fähigkeit, diese sinnvoll im Kontext neuer Probleme einzusetzen, zu analysieren und zu bewerten.		
Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.		
Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)		
Voraussetzungen: Empfehlenswert: Gutes Verständnis des Informatik III-Stoffes, insbesondere im Bereich der Graphenalgorithmen. Modul Informatik 3 (INF-0111) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Algorithmen für NP-harte Probleme (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 4		
Inhalte: NP-harte Probleme können nach heutigem Wissen nicht in polynomieller Zeit auf einem üblichen Rechner gelöst werden. Ungeachtet dessen treten solche Probleme überaus häufig in der Praxis auf, z.B. bei vielen Planungsaufgaben, und es ist von großer ökonomischer Bedeutung, sie doch noch zu lösen, zumindest "so gut wie es geht". Die Vorlesung behandelt Methoden der Algorithmentheorie, die hierfür entwickelt wurden. Einige Stichpunkte: Approximationsalgorithmen, Branch-and-Bound, Parametrisierung. Es werden auch Grenzen dieser Methoden aufgezeichnet.		
Literatur: <ul style="list-style-type: none"> • Skript • Ausiello et al., Complexity and Approximation, Springer, Berlin, 1999. • J. Hromkovic, Algorithmics for Hard Problems, Springer, Berlin, 2001. 		
Modulteil: Algorithmen für NP-harte Probleme (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 2		

Prüfung

Algorithmen für NP-harte Probleme (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0052: Einführung in die Komplexitätstheorie <i>Introduction to Complexity Theory</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
Lernziele/Kompetenzen: Verständnis für zentrale Themen und Methoden der Komplexitätstheorie; die Fähigkeit, komplexitätstheoretische Sachverhalte und Fragen präzise zu formulieren und zu untersuchen.		
Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.		
Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)		
Voraussetzungen: Empfehlenswert: Gutes Verständnis des Stoffes aus Einführung in die Theoretische Informatik sowie Informatik III, insbesondere bzgl. Turing-Maschinen und Graphenalgorithmen. Modul Informatik 3 (INF-0111) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Einführung in die Komplexitätstheorie (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2		
Inhalte: Aufbauend auf den in den Grundvorlesungen Einführung in die Theoretische Informatik und Informatik III gelegten Grundlagen werden wichtige Aspekte der Komplexitätstheorie behandelt. Das Anliegen der Komplexitätstheorie ist es, die inhärente Schwierigkeit von Berechnungsproblemen zu untersuchen und somit die prinzipiellen Grenzen effizienter Algorithmen zu beleuchten.		
Literatur: <ul style="list-style-type: none"> • Skript • Christos H. Papadimitriou, Computational Complexity, Addison-Wesley, Reading, Mass., 1994. 		
Modulteil: Einführung in die Komplexitätstheorie (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 2		

Prüfung

Einführung in die Komplexitätstheorie (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0053: I/O-effiziente Algorithmen <i>I/O-Efficient Algorithms</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
<p>Lernziele/Kompetenzen: Verständnis für den effizienten Umgang mit Speicherhierarchien; Kenntnis grundlegender I/O-effizienter Algorithmen, insbesondere für Sortieren und verwandte Probleme; Fähigkeit zur Analyse und Bewertung einfacher neuer Algorithmen im I/O-Modell; Verständnis für die Grenzen I/O-effizienter Algorithmen.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: Empfehlenswert: Gutes Verständnis des Informatik III-Stoffes. Modul Informatik 3 (INF-0111) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: I/O-effiziente Algorithmen (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Das klassische Berechnungsmodell der Random-Access-Machine (RAM) stößt zunehmend an seine Grenzen. Der Grund ist, dass moderne Rechner nicht über den "flachen" Speicher der RAM verfügen, bei dem alle Speicherzellen "gleichberechtigt" sind, sondern eine ausgefeilte Speicherhierarchie mit Caches, Hauptspeicher und Hintergrundspeicher(n) besitzen. Im Allgemeinen sind "näher am CPU" gelegene Speicher deutlich schneller, dafür aber kleiner, und ein effizienter Algorithmus muss versuchen, häufig benutzte Daten in Speicher mit kurzen Zugriffszeiten zu halten. In der Vorlesung werden wir uns, nach einer Einführung geeigneter Speichermodelle, aus theoretischer Sicht mit sogenannten I/O-effizienten oder "speicherbewussten" Algorithmen befassen, die die Anzahl der Datentransporte zwischen Stufen der Speicherhierarchie möglichst gering halten. Bereits für das Problem des Sortierens wird sich herausstellen, dass die "I/O-effiziente Welt" ganz anders aussieht als die "RAM-Welt".</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • Skript • J.S. Vitter, Algorithms and data structures for external memory, Foundations and Trends in Theoretical Computer Science 2 (2008), pp. 305-474 		

Modulteil: I/O-effiziente Algorithmen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

I/O-effiziente Algorithmen (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Beschreibung:

Die Prüfung findet zum Ende des Semesters statt. Es besteht bei Bedarf die Möglichkeit einer Nachholprüfung im Folgesemester.

Modul INF-0054: Datenstrukturen <i>Data Structures</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
Lernziele/Kompetenzen: Kenntnis einer Reihe von nichtelementaren Datenstrukturen und ihrer Analyse; Fähigkeit zur Anpassung dieser Datenstrukturen an neue Anwendungen und zur Entwicklung neuer einfacher Datenstrukturen.		
Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.		
Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)		
Voraussetzungen: Empfehlenswert: Gutes Verständnis des Informatik III-Stoffes. Modul Informatik 3 (INF-0111) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	

Moduleile
Modulteil: Datenstrukturen (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 4
Inhalte: Datenstrukturen realisieren abstrakte Datentypen so, dass die Operationen der Datentypen besonders effizient ausgeführt werden können. Beispiele von Datenstrukturen sind balancierte Bäume und Hashtabellen. Datenstrukturen können mit objektorientierten Programmiersprachen als Klassen zur Verfügung gestellt werden. In der Vorlesung werden verschiedene Datenstrukturen behandelt, die über die in Informatik III behandelten Datenstrukturen hinausgehen, unter anderem die sogenannten dynamischen Bäume von Sleator und Tarjan, Range-Query-Strukturen und Suffix-Bäume.
Literatur: Skript
Zugeordnete Lehrveranstaltungen: Datenstrukturen (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i> Datenstrukturen realisieren abstrakte Datentypen so, dass die Operationen der Datentypen besonders effizient ausgeführt werden können. Beispiele von Datenstrukturen sind balancierte Bäume und Hashtabellen. Datenstrukturen können mit objektorientierten Programmiersprachen als Klassen zur Verfügung gestellt werden. In der Vorlesung werden verschiedene Datenstrukturen behandelt, die über die in Informatik III behandelten

Datenstrukturen hinausgehen, unter anderem die sogenannten dynamischen Bäume von Sleator und Tarjan, Range-Query-Strukturen und Suffix-Bäume.

Modulteil: Datenstrukturen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Datenstrukturen (Übung)

Veranstaltung wird online/digital abgehalten.

Bitte melden Sie sich in der Digicampus-Veranstaltung der Vorlesung an.

Prüfung

Datenstrukturen (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0056: Online-Algorithmen <i>Online Algorithms</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
<p>Lernziele/Kompetenzen: Verständnis der Online-Problematik; Kenntnis fundamentaler Online-Probleme und -Algorithmen; Fähigkeit zum selbstständigen Entwurf einfacher Online-Algorithmen, zu ihrer kompetitiven Analyse mittels Potentialfunktionen und zu ihrer Bewertung.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Online-Algorithmen (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Manchmal muss man Entscheidungen treffen, bevor alle relevanten Daten bekannt sind. Will man z. B. Aktien kaufen, so wäre es sehr hilfreich, über die künftige Entwicklung aller Aktienkurse informiert zu sein; aber es liegt in der Natur der Sache, dass man den Kauf tätigen muss, bevor diese Information vorliegt. Ein zweites Beispiel: Eine Funktaxizentrale muss nach jeder Bestellung einen der verfügbaren Wagen auswählen und zum Fahrgast schicken; mit Wissen über später eintreffende Anrufe könnten die Wagen vielleicht günstiger auf die Fahrgäste verteilt werden. Algorithmen, die Entscheidungen bei unvollständiger Information treffen, heißen Online-Algorithmen. Die Vorlesung behandelt Online-Algorithmen und ihre Analyse.</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • Skript • A. Borodin und R. El-Yaniv, Online Computation and Competitive Analysis, Cambridge University Press, 1998. 		
Modulteil: Online-Algorithmen (Übung)		
Lehrformen: Übung		
Sprache: Deutsch		
SWS: 2		

Prüfung

Online-Algorithmen (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0066: Organic Computing II <i>Organic Computing II</i>		5 ECTS/LP
Version 1.3.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen: Teilnehmer sind in der Lage, wesentliche Konzepte und Methoden des Organic Computing (OC) wiederzugeben. Dazu zählen unter anderem: Selbst-Organisation, Selbst-Adaption, Robustheit, Flexibilität, Observer/Controller-Architekturen (O/C), Selbst-X Eigenschaften, Extended Classifier Systems (XCS), Genetische Algorithmen (GA), Partikel Swarm Optimization, Influence Detection und Trust. Sie können zudem begründen, wieso es sinnvoll ist OC-Systeme zu betrachten und praxisrelevante Beispiele nennen, in welchen OC-Techniken angewendet werden sollten. Die Teilnehmer sind befähigt, größere Softwaresysteme mit Hilfe der O/C-Architektur zu entwickeln, sowie diese mit passenden OC-Techniken zu befüllen und sich, falls erforderlich, neue OC-Techniken anzueignen ein gegebenes Problem zu lösen. Die Studenten haben die Fähigkeit zum Lösen von praktischen Aufgaben mit Hilfe des XCS, dem Swarming und einem GA. Sie können zudem ein Konzept für ein OC-System ausarbeiten und beurteilen, sowie wissenschaftliche Bewertungen im Bezug auf OC-Systeme durchführen.</p> <p>Schlüsselqualifikationen: Fähigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen, Fähigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten., Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen, Fertigkeit der Zusammenarbeit in Teams.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Organic Computing II (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Vorlesung "Organic Computing" vermittelt Ansätze zur Organisation von komplexen vernetzten Systemen, die aus einer Vielzahl von autonomen Teilsystemen bestehen. Dazu werden zunächst Anforderungen und Ziele solcher Systeme definiert und diskutiert. Darüber hinaus werden Konzepte aus dem Bereich der Systemarchitekturen und Ansätze aus dem Bereich naturanaloger Algorithmen dargestellt und bewertet. In allen Teilen werden Bezüge zu konkreten Anwendungsgebieten gegeben. Die zugehörige Übung bietet die Möglichkeit, die erlernten Ansätze zu vertiefen und beispielhaft anzuwenden.</p>		

Literatur:

- Folien
- Müller-Schloer und Tomforde: Organic Computing - Technical Systems for Survival in the Real World, Springer International Publishing, 2018, ISBN 978-3-319-68477-2
- Müller-Schloer et al.: Organic Computing - A Paradigm Shift for Complex Systems, Birkhäuser Verlag, Basel, 2011, ISBN 978-3034801294
- Würtz (ed.): Organic Computing (Understanding Complex Systems), SpringerVerlag Berlin, 2008, ISBN 978-3540776567
- Mitchell: Machine Learning, The McGraw-Hill Companies, 1997, ISBN 978-0071154673
- Goldberg: Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, 1989, ISBN 978-0201157673
- Michalewicz, Fogel: How to Solve it: Modern Heuristics, Springer Verlag Berlin, 2004, ISBN 978-3540224945
- Tomforde: Runtime Adaptation of Technical Systems, Südwestdeutscher Verlag für Hochschulschriften, 2012, ISBN 978-3838131337

Modulteil: Organic Computing II (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Organic Computing II (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Beschreibung:

Die Prüfung kann jedes Semester zu Beginn und Ende der vorlesungsfreien Zeit abgelegt werden.

Modul INF-0067: Peer-to-Peer und Cloud Computing <i>Peer-to-Peer and Cloud Computing</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen: Die erfolgreiche Teilnahme an dieser Lehrveranstaltung resultiert in fachlichen Kompetenzen in zwei Teilbereichen: Peer-to-Peer-Systeme und Cloud-Computing.</p> <p>Peer-to-Peer-Systeme Die Studierenden können verschiedene Peer-to-Peer-System-Architekturen und deren Vor- und Nachteile beschreiben. Insbesondere sind sie in der Lage, für einen gegebenen Anwendungsfall eine passende Architektur auszuwählen und diese Entscheidung zu begründen. Des Weiteren können sie gängige dezentrale Suchalgorithmen und -heuristiken einordnen und bezüglich ihrer Effizienz vergleichen. Sie verstehen abstrakte Netzwerkmodelle, und können mit deren Hilfe Peer-to-Peer-Netzwerke analysieren.</p> <p>Cloud-Computing Teilnehmer können die dem Cloud-Computing zugrunde liegenden Technologien unterscheiden und basierend auf diesem Wissen Cloud-Computing-Architekturen analysieren. Sie kennen die Wichtigkeit von Lastverteilung im Rahmen von großen Cloud-Computing-Anwendungen und können die verschiedenen Ansätze dafür erklären und gegeneinander abwägen. Die Studierenden kennen moderne Ansätze zu nebenläufigen Berechnungen über große Datenmengen und können diese auf einfache Probleme anwenden. Sie sind in der Lage, einfache Anwendungen mithilfe von Platform-as-a-Service-Frameworks zu entwickeln.</p> <p>Schlüsselqualifikationen Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Modulteil: Peer-to-Peer und Cloud Computing (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		

Inhalte:

Dieses Modul vermittelt Ansätze zur Organisation von komplexen abstrahierten IT-Infrastrukturen, die dynamisch an wechselnde Nutzungsbedingungen angepasst werden können und Dienste wie Rechenkapazität, Datenspeicher, Netzkapazitäten oder Software auf verschiedenen Ebenen zur Verfügung stellen. Dazu werden die Anforderungen, Eigenschaften und Ziele solcher Systeme definiert und diskutiert. Darüber hinaus werden Konzepte aus dem Bereich der Systemarchitekturen und Ansätze aus dem Bereich selbstorganisierender Algorithmen dargestellt und bewertet. In allen Teilen werden Bezüge zu konkreten Anwendungsgebieten gegeben.

Literatur:

- aktuelle wissenschaftliche Paper
- Mahlmann und Schindelhauer: Peer-to-Peer Netzwerke - Algorithmen und Methoden, Springer 2007
- Antonopoulos und Gillam: Cloud Computing - Principles, Systems and Applications, Springer 2010

Modulteil: Peer-to-Peer und Cloud Computing (Übung)

Lehrformen: Übung

Sprache: Deutsch

Angebotshäufigkeit: unregelmäßig (i. d. R. im WS)

SWS: 2

Prüfung

Peer-to-Peer und Cloud Computing (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Beschreibung:

Die Prüfung kann jedes Semester zu Beginn und Ende der vorlesungsfreien Zeit abgelegt werden.

Modul INF-0069: Weiterführende Betriebssystemkonzepte <i>Advanced Operating System Concepts</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen: Die Veranstaltung vermittelt den Studierenden bei erfolgreicher Teilnahme fundierte fachliche Kenntnisse über moderne Betriebssysteme und die dahinterstehenden grundlegenden Konzepte. Sie können danach die Geschichte der Betriebssysteme wiedergeben und deren primäre Typen erläutern. Der theoretische Teil konzentriert sich auf die Vertiefung des Verständnisses von Mechanismen aus den Bereichen Multiprogramming, Interprozesskommunikation, Speicher-Management und Virtualisierung. Die Studierenden können die kennengelernten Ansätze hinsichtlich ihrer Eignung für einen bestimmten Einsatzzweck beurteilen. Der praktische Teil befasst sich mit dem Transfer und der Anwendung des im theoretischen Teil vermittelten Wissens. Sie benutzen Programme und Programmierschnittstellen aus dem Unix-Umfeld, um ein erarbeitetes Lösungskonzept mit dem Schwerpunkt Ein-/Ausgabe durch einen Programmcode zu realisieren.</p> <p>Schlüsselqualifikationen: Qualitätsbewusstsein, Akribie; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Kenntnisse der Vor-/Nachteile von Entwurfsalternativen, Bewertung im jeweiligen Anwendungszusammenhang; Fachspezifische Vertiefungen; Umsetzen fachlicher Lösungskonzepte in Programme und Modelle.</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vorlesung (Präsenzstudium) 60 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Moduleil: Weiterführende Betriebssystemkonzepte (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		

Inhalte:

Die Vorlesung beinhaltet u. a. folgende Themen:

- Grundlagen eines Betriebssystems
 - Aufgaben
 - Struktur und Aufbau
 - Historie
 - Shell
- Prozesse, Threads und Interrupts
 - Prozesse und Threads
 - HW-Interrupts
- Scheduling
 - Grundlagen des Scheduling
 - Echtzeit-Scheduling
- Speicher
 - Interaktion
 - Speicherkonzepte
 - Dateisysteme
- Virtualisierung

Literatur:

- Folien
- Andrew S. Tanenbaum: "Moderne Betriebssysteme", Pearson Studium, ISBN:978-3-8273-7342-7

Modulteil: Weiterführende Betriebssystemkonzepte (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Prüfung

Weiterführende Betriebssystemkonzepte

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Beschreibung:

Die Prüfung kann jedes Semester zu Beginn und Ende der vorlesungsfreien Zeit abgelegt werden.

Modul INF-0077: Suchmaschinen <i>Search Engines</i>		8 ECTS/LP
Version 1.2.0 (seit SoSe14) Modulverantwortliche/r: PD Dr. Markus Endres		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an der Veranstaltung sind die Studierenden in der Lage die Konzepte und Methoden, Verfahren, Techniken und Technologien von Suchmaschinen zu verstehen und zu bewerten. Hierfür sind vertiefende Kenntnisse im Bereich der Suchmaschinentechologie notwendig. Darunter zählen z.B. die Architektur einer Suchmaschine, Web Crawler, Textprocessing, Indexierung, verschiedene Retrieval Modelle, die Evaluation von Suchalgorithmen, Clusterverfahren, Link Analyse, uvm.</p> <p>Die Studierenden erwerben somit Kenntnisse verschiedener Suchmaschinenaspekte und können die Vor- und Nachteile in praxisrelevanten Aufgabenstellungen abwägen und beurteilen. Zudem können die Studierenden nach der Veranstaltung wissenschaftlich aussagekräftige Bewertungen erstellen und diese in praxisrelevante Lösungsstrategien umsetzen. In der Veranstaltung werden zudem relevante Themen aus der Forschung anhand aktueller Veröffentlichungen angesprochen, so dass die Studierenden die Fähigkeit erwerben, ihr vorhandenes Wissen selbständig zu erweitern.</p>		
<p>Schlüsselqualifikationen: Fachspezifische Vertiefung; Umsetzen fachlicher Lösungskonzepte in Programme und Modelle; Kenntnisse der Vor-/Nachteile von Entwurfsalternativen und Bewertung im jeweiligen Anwendungszusammenhang; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit, mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen anzustellen; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen;</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Modul Datenbanksysteme (INF-0073) - empfohlen		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Suchmaschinen (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 4		
Inhalte: Die Vorlesung behandelt grundlegende Konzepte von Suchmaschinen, Volltext-Suche, SQL-Suchmaschinen und Präferenz-Suchmaschinen (Preference SQL) sowie deren Implementierung.		

Literatur:

- M. Levene: An Introduction to Search Engines and Web Navigation
- R. Baeza-Yates, B. Ribeiro-Neto: Modern Information Retrieval
- I. H. Witten, M. Gori, T. Numerico: Web Dragons
- W. Kießling: Foundations of Preferences in Database Systems
- W. Kießling: Preference Queries with SV-Semantics

Modulteil: Suchmaschinen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Suchmaschinen (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0078: Datenbankprogrammierung (Oracle) <i>Database Programming (Oracle)</i>		5 ECTS/LP
Version 1.2.0 (seit SoSe14) Modulverantwortliche/r: PD Dr. Markus Endres		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an der Veranstaltung sind die Studierenden in der Lage vertiefte Oracle-Kenntnisse in praxisrelevanten Aufgaben anzuwenden. Hierzu ist erweitertes Wissen im Bereich DB-System-Architektur, DB-Entwurfstheorie, Administration, erweitertes SQL, PL/SQL-Programmierung, Baumstrukturen, XML und OLAP notwendig. Die Studierenden erlangen eine fachspezifische Vertiefung im Bereich der Datenbanksysteme und somit die Fertigkeit zur Analyse und Strukturierung von Problemstellungen. Darüber hinaus können die Studierenden diese komplexen, praxisrelevanten Aufgabenstellungen auf dem Gebiet Datenbanken, insbesondere unter Verwendung von Oracle, analysieren, bewerten und lösen. Sie kennen die Vor- und Nachteile unterschiedlicher ER-Modellierungen und können durch logisches und konzeptionelles Denken eine geeignete Lösung für komplexe Problemstellungen unter praxisnahen Randbedingungen verfassen.</p> <p>Schlüsselqualifikationen: Umsetzen fachlicher Lösungskonzepte in Programm und Modelle, Fachspezifische Vertiefung; Fachübergreifende Kenntnisse; Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen; Kenntnisse der Vor-/Nachteile von Entwurfalternativen und Bewertung im jeweiligen Zusammenhang; Auswahl und sichere Anwendung geeigneter Methoden; Kenntnisse von praxisrelevanten Aufgabenstellungen; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen;</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Modul Datenbanksysteme (INF-0073) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Datenbankprogrammierung (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch / Englisch		
SWS: 2		
<p>Inhalte: Die Vorlesung behandelt Problemlösungsstrategien unter Zuhilfenahme einer Oracle-Datenbank. Dazu werden die Oracle-Architektur, Zugriffsrechte, Transformation von ER nach SQL, Oracle SQL, Aktive Inhalte wie PL/SQL und Java in Oracle, XML-Unterstützung in Oracle, Baumstrukturen, Tuning, Backup und Recovery behandelt.</p>		

Literatur:

- R. Elmasri, S. Navathe: Fundamentals of Database Systems
- S. Melton: Understanding the New SQL: A Complete Guide
- Oracle 11g Online-Dokumentation

Zugeordnete Lehrveranstaltungen:

Datenbankprogrammierung (Oracle) (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Datenbanken haben sich als allgegenwärtiges Werkzeug im öffentlichen, wissenschaftlichen und wirtschaftlichen Leben etabliert. Diese Vorlesung wendet sich an DB-Interessierte, die ihre vorhandenen Kenntnisse aus einer grundlegenden Datenbankvorlesung mit Hilfe von Oracle vertiefen bzw. erweitern wollen. Daher ist die Vorlesung insbesondere für Studierende geeignet, die ihren Schwerpunkt im Bereich Datenbanken- und Informationssysteme setzen bzw. vertiefte praktische Kenntnisse erwerben wollen.

Modulteil: Datenbankprogrammierung (Übung)

Lehrformen: Übung

Sprache: Deutsch / Englisch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Datenbankprogrammierung (Oracle) (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Datenbankprogrammierung (Klausur)

Klausur / Prüfungsdauer: 60 Minuten

Modul INF-0088: Bayesian Networks <i>Bayesian Networks</i>		5 ECTS/LP
Version 1.0.0 (seit WS15/16) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Bayessche Netze gehören zu den vielseitigsten statistischen Methoden des maschinellen Lernens. Nach einer erfolgreichen Teilnahme an diesem Modul verstehen und vertiefen Teilnehmer die Kernprinzipien von Bayesschen Netzen und können diese auf viele praxisrelevante Probleme in unterschiedlichen Fachgebieten anwenden. Diese umfassen unter anderem die Robotik, Websuche, intelligente Agenten, automatisierte Diagnosesysteme und medizinische Systeme. Die Studierenden sind in der Lage, Bayessche Netze zu verstehen, anzuwenden und fachübergreifende Problemstellungen in diesem Kontext zu analysieren und zu evaluieren. Die Teilnahme an diesem Modul fördert Fertigkeiten zum logischen, analytischen und konzeptionellen Denken. Studierende sind in der Lage, mittels Bayesscher Netze wissenschaftlich aussagekräftige Bewertungen zu erstellen.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Logik; umsetzen fachlicher Lösungskonzepte in Modelle; fachübergreifende Kenntnisse; Entwicklung und Umsetzung von Lösungsstrategien komplexer Problemstellungen; systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen</p>		
<p>Bemerkung: Die gleichzeitige Einbringung von diesem Modul und INF-0263 ist nicht möglich.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen: keine</p>		
<p>Angebotshäufigkeit: jedes Sommersemester</p>	<p>Empfohlenes Fachsemester: ab dem 5.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 4</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p>		
<p>Modulteil: Bayesian Networks (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		
<p>Inhalte:</p> <ol style="list-style-type: none"> 1. Basics of Probability Theory 2. Example: Bayesian Network based Face Detection 3. Inference 4. Influence Diagrams 5. Parameter Learning 6. Example: probabilistic Latent Semantic Analysis (pLSA) 		

Literatur:

- Richard E. Neapolitan. Learning Bayesian Networks. Prentice Hall Series in Artificial Intelligence, 2004. ISBN 0-13-012534-2
- Daphne Koller, Nir Friedman. Probabilistic Graphical Models: Principles and Techniques. The MIT Press, 2009. 978-0262013192

Modulteil: Bayesian Networks (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Bayesian Networks (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Beschreibung:

Die Prüfung kann jedes Semester in der Prüfungszeit abgelegt werden.

Modul INF-0093: Probabilistic Robotics <i>Probabilistic Robotics</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach einer erfolgreichen Teilnahme an diesem Modul verstehen Teilnehmer Grundlagen und vertiefende Fragestellungen und Algorithmen der Robotik (z.B. rekursive Zustandsschätzung, gaußsche- und nicht-parametrische Filter, Kalman Filter, Bewegung und Lokalisierung, Perzeption, Kartierung, SLAM) aus wahrscheinlichkeitstheoretischer Sicht und können erlernte Konzepte auf komplexe, praxisrelevante Aufgabenstellung anwenden. Die Studierenden sind in der Lage, Problemstellungen in diesem Kontext zu analysieren und zu evaluieren. Die Teilnahme an diesem Modul fördert Fertigkeiten zum logischen, analytischen und konzeptionellen Denken auf dem Gebiet der wahrscheinlichkeitstheoretischen Robotik. Studierende können aus den erlernten Konzepten zielgerichtet geeignete Methoden auswählen, sicher anwenden und auf neue, auch fachfremde, Fragestellungen übertragen. Das Modul vermittelt Kompetenzen zum Erkennen aktueller Forschung und bedeutenden technischen Entwicklungen auf diesem Gebiet.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Logik; umsetzen fachlicher Lösungskonzepte in Modelle; fachübergreifende Kenntnisse; Entwicklung und Umsetzung von Lösungsstrategien komplexer Problemstellungen; systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Probabilistic Robotics (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch Angebotshäufigkeit: jedes Wintersemester SWS: 2</p>		

Inhalte:

1. Introduction to Probabilistic Robotics
2. Recursive State Estimation
3. Gaussian Filters
4. Nonparametric Filters
5. Robot Motion
6. Robot Perception
7. Mobile Robot Localization: Markow and Gaussian
8. Mobile Robot Localization: Grid and MonteCarlo
9. Occupancy Grid Mapping
10. SLAM

Literatur:

Sebastian Thrun, Wolfram Burgard, Dieter Fox. Probabilistic Robotics. Springer Verlag.

Modulteil: Probabilistic Robotics (Übung)

Lehrformen: Übung

Sprache: Deutsch

Angebotshäufigkeit: jedes Wintersemester

SWS: 2

Prüfung

Probabilistic Robotics (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Beschreibung:

Die Prüfung kann jedes Semester in der Prüfungszeit abgelegt werden.

Modul INF-0129: Softwaretechnik II <i>Software Engineering II</i>		8 ECTS/LP
Version 2.0.0 (seit WS16/17) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen:</p> <p>Fachbezogene Kompetenzen:</p> <p>Nach dem erfolgreichen Besuch des Moduls sind die Teilnehmer in der Lage, fortgeschrittene Verfahren der agilen Softwareentwicklung, des Requirements Engineerings, des Testens und des Refactorings von Softwaresystemen anzuwenden und miteinander zu verbinden. Sie können die Anwendbarkeit der verschiedenen Verfahren und Methoden situationsspezifisch beurteilen.</p> <p>Die Teilnehmer verstehen die Aufgaben des Requirements Engineering und sind in der Lage, wesentliche Methoden der Requirements-Erfassung und Dokumentation (Use-Cases, Satzschablonen, formale Modelle) anzuwenden und die Eignung der verschiedenen Dokumentationsformen im Projektkontext zu bewerten. Sie haben die Fertigkeit zum analytischen und konzeptionellen Denken und können so systematisch komplexe Kundenanforderungen analysieren und können diese formulieren und in geeigneten Modellen darstellen.</p> <p>Die Studierenden verstehen einen agilen Entwicklungsprozess und die Basisprinzipien agiler Vorgehensweisen. Sie können den agilen Entwicklungsprozess auf Projekte übertragen und seine Anwendbarkeit im Projektkontext beurteilen. Sie kennen agile Praktiken und deren Wechselwirkungen und können die Praktiken einsetzen. Sie verstehen die Aufgaben der beteiligten Rollen und die Bausteine des Entwicklungsprozesses.</p> <p>Die Studierenden verstehen die Aufgaben des Testens und die Teilschritte des Testprozesses. Sie kennen Verfahren zur Bestimmung von Testfällen und zur Bewertung der Adäquatheit von Testsuiten und können diese anwenden.</p> <p>Die Studierenden kennen die Prinzipien guten objektorientierten Designs und können durch Refactoring die Qualität des Designs eines Systems verbessern. Sie können Ideen und Konzepte sicher und überzeugend darstellen und haben die Fähigkeit zur Zusammenarbeit im Team.</p> <p>Die Studierenden erwerben insgesamt Kenntnisse in Methoden zur Entwicklung großer Softwaresysteme und der Konstruktion von Abstraktionen sowie die Fähigkeit zu deren Anwendung im Projektkontext. Sie können alternative Vorgehensweisen wissenschaftlich fundiert bewerten und für ein Projekt geeignete auswählen.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Abstraktionsfähigkeit • Moderieren fachlicher Sitzungen • Erlernen des eigenständigen Arbeitens mit Lehrbüchern und Arbeit in selbstorganisierten Teams • Kenntnisse von praxisrelevanten Aufgabenstellungen 		
<p>Arbeitsaufwand:</p> <p>Gesamt: 240 Std.</p> <p>90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>30 Std. Übung (Präsenzstudium)</p> <p>60 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen:</p> <p>Programmierkenntnisse in Java (empfohlen)</p> <p>Modul Softwaretechnik (INF-0120) - empfohlen</p>		
<p>Angebotshäufigkeit: jedes Wintersemester</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>

SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
------------------	---	--

Moduleile**Modulteil: Softwaretechnik II (Vorlesung)****Lehrformen:** Vorlesung**Sprache:** Deutsch**SWS:** 2**Inhalte:****Agile Softwareentwicklung:**

- Entwicklungsmethoden (Scrum)
- Agile Praktiken
- Agile Werte, Prinzipien und Methoden

Refactoring

- Code Smells
- Prinzipien des objektorientierten Designs
- Wichtige Refactorings

Testen

- Testprozess und Ziele des Testens
- Testarten
- Methoden zur Testfallgewinnung
- Adäquatheitskriterien beim Blackbox- und Whitebox-Testen

Requirements Engineering

- Aufgaben, Begriffe und Artefakte
- RE-Prozess
- Techniken zur Requirements-Elicitation, -Analyse und -Dokumentation
- Qualitätskriterien für Software-Requirements

Literatur:

- Pohl, Rupp: Basiswissen Requirements Engineering, dpunkt Verlag 2009
- U. Hammerschall, G. Benekean: Software Requirements, Pearson 2013
- S. Robertson, J. Robertson: Mastering the Requirements Process, Addison-Wesley 2013
- Bleek, Wolf: Agile Softwareentwicklung, dpunkt Verlag 2008
- R. Pichler: Scrum, dpunkt.verlag 2008
- Spillner, Linz: Basiswissen Softwaretest, dpunkt Verlag 2005
- Fowler: Refactoring, Addison-Wesley 1999
- Vorlesungsfolien mit schriftlichen Ergänzungen und Anmerkungen

Zugeordnete Lehrveranstaltungen:**Softwaretechnik 2 (Vorlesung)**

Veranstaltung wird als Hybrid/gemischt abgehalten.

Die Veranstaltung findet im Wesentlichen digital statt. Vorlesungsinhalte werden als Screencast zum Download angeboten. Zusätzlich gibt es pro Woche höchstens einen synchronen Termin, bei dem Fragen zum Stoff gestellt werden können und die Übungen behandelt werden. Für den synchronen Termin ist im Stundenplan Dienstag ab 16 Uhr vorgesehen. Falls es die Corona-Situation und die Teilnehmerzahl erlauben, gibt es am Anfang des neuen Jahres noch zwei Präsenztermine, in denen Stoff praktisch erarbeitet wird.

Modulteil: Softwaretechnik II (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Zugeordnete Lehrveranstaltungen:

Übung zu Softwaretechnik 2 (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Softwaretechnik II Klausur

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0130: Formale Methoden im Software Engineering <i>Formal Methods in Software Engineering</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
Lernziele/Kompetenzen: Fachbezogene Kompetenzen: <p>Die Studierenden können mathematisch-formale Methoden für die Programmverifikation, speziell bei sicherheitskritischer Software einsetzen. Durch das Verständnis von Kalkülen trainieren sie die Fertigkeit zum logischen, konzeptuellen und analytischen Denken. Sie können Spezifikationen von Datenstrukturen entwerfen und deren Eigenschaften formal beweisen. Sie sind in der Lage, funktionale Eigenschaften von sequentiellen und nebenläufigen Programmen zu formulieren und dafür Beweise zu erstellen. Sie erwerben die Fertigkeit zur abstrakten, modularen Modellierung von Softwaresystemen und zur systematischen Entwicklung von korrekten Programmen aus diesen Modellen. Sie erwerben dadurch die Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen und können mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen abgeben. Sie erwerben Fähigkeiten, um Beiträge zur Wissenschaft zu leisten.</p> Schlüsselqualifikationen: <ul style="list-style-type: none"> • Training des logischen Denkens • Analytisch-methodische Kompetenz • Erwerb von Abstraktionsfähigkeiten • Fähigkeiten zur Modellbildung für Softwaresysteme 		
Arbeitsaufwand: Gesamt: 240 Std. 60 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Formale Methoden im Software Engineering (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
Inhalte: <p>Übergeordnetes Ziel ist die Produktion beweisbar korrekter Software. In der Vorlesung werden verschiedene klassische Methoden für die Programmverifikation im Kleinen behandelt. Darüber hinaus werde innovative Techniken für die formale Modellierung und Verifikation großer Systeme vermittelt. Als Werkzeug kommt das KIV-System zum Einsatz, das die formale Spezifikation und Verifikation von Systemen ermöglicht. Konkrete Inhalte sind: Algebraische Spezifikationen, interaktives Theorembeweisen, Hoare-Logik, Dynamische Logik, Temporallogik</p>		

Literatur:

- Sperschneider, Antoniou: Logic: A Foundation for Computer Science, Addison Wesley 1991
- Loeckx, Ehrich, Wolf: Specification of Abstract Data Types, Wiley 1996
- Ausführliche Dokumentation
- Folienhandout

Modulteil: Formale Methoden im Software Engineering (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Prüfung

Formale Methoden im Software Engineering (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 45 Minuten

Modul INF-0131: Software- und Systemsicherheit <i>Software and Systems Security</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen:</p> <p>Fachbezogene Kompetenzen:</p> <p>Nach der erfolgreichen Teilnahme an der Lehrveranstaltung kennen Studierende die Grundlagen, Aufbau, und typische Schwächen kryptographischer Protokolle und praxisrelevanter, sicherheitskritischer Systeme, können diese verständlich und sicher darstellen, und verstehen die Grundprinzipien deren Sicherheitsanalyse.</p> <p>Sie können Bedrohungsanalyse durchführen, kryptographische Protokolle entwickeln, Chipkarten programmieren, sicherheitskritische Systeme in Teams entwerfen, und ihre Ergebnisse dokumentieren.</p> <p>Sie können systematisch Bedrohungen für Softwaresysteme analysieren und deren Risiken bewerten.</p> <p>Sie können einen modellgetriebenen Entwicklungsprozess für sicherheitskritische Systeme anwenden.</p> <p>Sie sind in der Lage, bedeutende technische Entwicklungen zu erkennen und fachliche Lösungskonzepte in Programme umzusetzen. Sie haben Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Abstraktionsfähigkeiten • Grundsätze guter wissenschaftlicher Praxis • Mathematisch-formale Grundlagen • Quantitative Aspekte der Informatik • Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen • Fähigkeit, vorhandenes Wissen selbstständig zu erweitern • Qualitätsbewusstsein und Akribie 		
<p>Arbeitsaufwand:</p> <p>Gesamt: 240 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>60 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig (i. d. R. im WS)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Software- und Systemsicherheit (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Deutsch</p> <p>SWS: 2</p>		

Inhalte:

Inhalt der Vorlesung ist der Entwurf sicherer Softwaresysteme, speziell verteilter Systeme, deren Sicherheit wesentlich auf dem Einsatz von Sicherheitsprotokollen beruht. Ein Schwerpunkt liegt dabei auf Anwendungen, in denen Chipkarten eingesetzt werden. In der Vorlesung werden Kenntnisse in JavaCard, der Chipkartentechnologie, Bedrohungsanalyse und dem Design kryptographischer Anwendungsprotokolle vermittelt, die in den Übungen an praktischen Beispielen (u.a. einer elektronischen Kopierkarte und einer elektronischen Fahrkarte) erprobt werden. Bei der Entwicklung der Protokolle wird der SecureMDD-Ansatz verwendet, eine Methode zur modellgetriebenen Entwicklung sicherheitskritischer Protokolle.

Literatur:

- Schneier: Applied Cryptography, Wiley and Sons, 1996 (2nd edition)
- Anderson, Needham: Programming Satan's Computer, in: Computer Science Today, Springer LNCS 1000, 1995
- Lowe: Breaking and fixing the Needham-Schroeder public-key protocol using FDR, in: Tools and Algorithms for the Construction and Analysis of Systems, Springer LNCS 1055, 1996
- Folienhandout, Spezifikationen und APIs

Modulteil: Software- und Systemsicherheit (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Prüfung

Software- und Systemsicherheit (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 45 Minuten

Modul INF-0133: Selbstorganisierende, adaptive Systeme <i>Self-organizing, adaptive systems</i>		8 ECTS/LP
Version 3.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Die Studierenden haben vertiefte Fachkenntnisse über die Eigenschaften und den Aufbau selbst-organisierender Systeme aus der Biologie, Soziologie, Physik und anderen Bereichen und der systematischen Modellierung und Konstruktion adaptiver Systeme in der Informatik und können solche Systeme analysieren und selbst entwerfen. Sie kennen Vor- und Nachteile verschiedener Entwurfsalternativen und können sie im Kontext der Problemstellung bewerten. Sie haben die Fertigkeit zum analytischen und konzeptionellen Denken und können geeignete Methoden auswählen und anwenden und wissenschaftlich aussagekräftige Bewertungen abgeben. Die Studierenden sind in der Lage, adaptive Systeme adäquat zu modellieren und dokumentieren.</p> <p>Außerdem kennen die Studierenden praxisrelevante Fragestellungen.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Erwerb von Abstraktionsfähigkeiten • Grundsätze guter wissenschaftlicher Praxis 		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vorlesung (Präsenzstudium) 60 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Selbstorganisierende, adaptive Systeme (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: In der Vorlesung werden die Grundlagen verschiedener Selbst-Organisationsmechanismen sowie das Handwerkszeug, um diese in IT-Systemen einsetzen zu können, vermittelt. Im Verlauf der Veranstaltung werden verschiedene Beispiele für selbstorganisierende Systeme vorgestellt, untersucht und Anwendungen der erlernten Organisationsprinzipien auf Beispiele aus der Informatik erläutert. Schließlich werden Methoden betrachtet, mit deren Hilfe sich Selbst-Organisation und Adaptivität in die Entwicklung komplexer Computersysteme integrieren lassen. Konkrete Themen sind: Selbst-Organisation, Emergenz, Chaostheorie, zelluläre Automaten, Spieltheorie, Multi-Agentensysteme, Autonomic Computing, Organic Computing.</p>		

Literatur:

- Gleick: Chaos: Making a New Science, Penguin 2008
- Strogatz: Sync : The Emerging Science of Spontaneous Order, Hyperion 2003
- Miller, Page: Complex Adaptive Systems: An Introduction to Computational Models of Social Life, Princeton University Press 2007
- Dawkins: The Selfish Gene, Oxford University Press, 3rd Revised Edition
- Wolfram: A New Kind of Science, Wolfram Media Inc. 2002
- von Neumann, Morgenstern: Theory of Games and Economic Behavior, 2004
- Folienhandout

Zugeordnete Lehrveranstaltungen:

Selbst-organisierende, adaptive Systeme (Vorlesung)

Veranstaltung wird online/digital abgehalten.

"Selbst-organisierende, adaptive Systeme" beschäftigt sich mit theoretischen Grundlagen, die für die Entwicklung von offenen Multiagentensystemen nötig sind. Dabei folgt die Vorlesung vor allem dem Aufbau von <http://www.masfoundations.org/> und behandelt Spieltheorie, Mechanism Design und (verteilte) Constraint-Optimierung. Sie richtet sich vor allem an den Anwendungsfällen des Lehrstuhls "Verteiltes Energiemanagement" sowie flexible Produktion (<https://www.uni-augsburg.de/de/fakultaet/fai/isse/projects/soproduction/>) und Planung und Selbstorganisation in mobilen Multi-Roboter-Systemen (<https://www.uni-augsburg.de/de/fakultaet/fai/isse/projects/combo/>) aus.

Modulteil: Selbstorganisierende, adaptive Systeme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Zugeordnete Lehrveranstaltungen:

Übung zu Selbst-organisierende, adaptive Systeme (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Selbstorganisierende, adaptive Systeme

Mündliche Prüfung / Prüfungsdauer: 45 Minuten

Beschreibung:

Ausnahmefall WiSe 2020/2021: Die Prüfung findet als Klausur statt.

Modul INF-0147: Prozessorarchitektur <i>Processor Architecture</i>		5 ECTS/LP
Version 2.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer Dr. Martin Frieb		
<p>Lernziele/Kompetenzen: Die Studierenden verstehen die Prinzipien des Aufbaus folgender Prozessoren auf einem fortgeschrittenen, praxisorientierten, aber wissenschaftlichen Niveau: superskalare Mikroprozessoren, Multi- und Manycore-Prozessoren, Signalprozessoren, Smartphone-Prozessoren und GPUs. Außerdem lernen sie aktuelle Konzepte der Prozessorarchitektur kennen und können Vor-/Nachteile aktueller und zukünftiger Prozessoren anhand ihres internen Aufbaus einschätzen und beurteilen. Sie verstehen die Funktionsweise von verschiedenen Komponenten von Mikroprozessoren und deren Zusammenspiel. Somit ist es den Studierenden möglich, den Einfluss verschiedener Architekturerweiterungen auf das Gesamtsystem einzuordnen. Ebenso können sie unterscheiden, wann es sinnvoll ist, speichergekoppelte, nachrichtengekoppelte oder datenparallele Prozessoren einzusetzen, und differenzieren zwischen Techniken für energieeffiziente oder hochperformante Prozessoren. In der Übung simulieren die Studierenden verschiedene Prozessorarchitekturen und Sprungvorhersagetechniken und bewerten deren Einfluss auf die Laufzeit und den Energieverbrauch.</p> <p>Schlüsselqualifikationen: Analytisch-methodische Kompetenz, Abwägung von Lösungsansätzen, Präsentation von Lösungen von Übungsaufgaben; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein, Akribie; Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen; Fertigkeit, mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen anzustellen; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Modul Systemnahe Informatik (INF-0138) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Prozessorarchitektur (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Vorlesung "Prozessorarchitektur" vertieft die Techniken superskalärer Mikroprozessoren und aktueller Multicore-Prozessoren. Dabei werden die Pipelinestufen detailliert behandelt, mehrfädige Prozessoren und Multicores gegenübergestellt sowie aktuelle Beispielprozessoren vorgestellt. Außerdem wird aus der Forschung an Manycores und Echtzeit-Multicores berichtet.</p>		

Literatur:

- Uwe Brinkschulte, Theo Ungerer, Mikrocontroller und Mikroprozessoren, Springer Verlag, Heidelberg, dritte Auflage 2010
- John L. Hennessy, David A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 5. Auflage, 2011

Modulteil: Prozessorarchitektur (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Prozessorarchitektur

Klausur / Prüfungsdauer: 60 Minuten

Modul INF-0149: Praktikum Eingebettete Systeme <i>Practical Module Embedded Systems</i>		5 ECTS/LP
Version 2.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen: Die Studierenden erwerben Kompetenzen in den folgenden Bereichen auf einem fortgeschrittenen, praxisorientierten, aber wissenschaftlichen Niveau: selbstständiges Arbeiten mit Mikrocontrollern, Datenblättern und Spezifikationen, Anbindung von analoger und digitaler Peripherie, Entwurf und Modellierung von eingebetteter Software mit Zustandsdiagrammen sowie deren Transformation in Code. Weitere Schwerpunkte sind die Konfiguration von sequentiellen Schnittstellen sowie Scheduling und task-basierte Programmierung.</p> <p>Durch die Implementierung von Aufgaben für einen Mikrocontroller setzen sie die erlernten Konzepte direkt in die Praxis um. Dabei steht Interaktion mit Sensoren und Aktoren sowie die Kommunikation mit anderen Systemteilen im Vordergrund. Dazu können sie unterschiedliche Arten der Ablaufsteuerung unterscheiden und anwenden. In einer abschließenden Projektphase lernen die Studierenden, eine komplexe Aufgabenstellung im Team zu planen, nach einem selbst entwickelten fundierten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zur verständlichen Darstellung und Dokumentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein, Akribie; Projektgebundene Arbeit und Zeitmanagement; Auswahl und sichere Anwendung geeigneter Methoden; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Selbstreflexion</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Praktikum (Präsenzstudium)</p>		
Voraussetzungen: Kenntnisse in C.		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Eingebettete Systeme		
Lehrformen: Praktikum		
Sprache: Deutsch		
SWS: 4		
<p>Inhalte:</p> <p>In dem Praktikum "Eingebettete Systeme" sollen die Einschränkungen und Herausforderungen für das Programmieren von eingebetteten Systemen erlernt werden. Als Plattform dient ein Staubsaugerroboter (ROOMBA) und ein daran angeschlossener Mikrocontroller zur Steuerung des ROOMBA. Die Programmierung erfolgt hardwarenah in C und die erstellten Programme sollen die Sensoren des ROOMBA auslesen und entsprechende Aktuatoren stellen. Dabei sollen insbesondere die Herausforderungen eingebetteter Systeme, wie z.B. das Zeitverhalten der Software sowie das Arbeiten mit Datenblättern kennengelernt werden.</p> <p>In einer Projektphase sollen die anfänglich erlernten Grundkenntnisse vertieft werden und komplexere Steuerungsprogramme entwickelt werden, z.B. ein autonomer Explorator oder ein "ROOMBA-Rennen" durch ein Labyrinth. Die Projekte werden einzeln oder im Team bearbeitet, dokumentiert und am Ende des Praktikums präsentiert.</p>		

Literatur:

- Marwedel, Wehmeyer: Eingebettete Systeme, Springer Verlag, Heidelberg, 2007
- Wiegmann: Softwareentwicklung in C für Mikroprozessoren und Mikrocontroller, 6. Auflage, VDE Verlag, Berlin, 2011

Prüfung

Praktikum Eingebettete Systeme

praktische Prüfung

Modul INF-0150: Hardware-Entwurf <i>Hardware Design</i>		8 ECTS/LP
Version 2.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer Dr. Martin Frieb		
<p>Lernziele/Kompetenzen: Die Studierenden erwerben Kompetenzen in den folgenden Bereichen auf einem fortgeschrittenen, praxisorientierten, aber wissenschaftlichen Niveau: Entwurfsprozess für digitale Schaltungen, Schaltungslogik und Gatter, physikalische Grundlagen der elektronischen Bauteile, Beschreibung von Hardware mit einer Hardwarebeschreibungssprache. Zunächst lernen die Studierenden die Verknüpfung logischer Gatter und bauen einen Halb- und einen Volladdierer. Sie verstehen den Entwicklungsprozess für digitale Schaltungen und wenden diesen direkt praktisch an, indem sie ihren eigenen RISC-V-Prozessor entwickeln. Sie modellieren und implementieren ihn selbstständig mit Hilfe der Hardwarebeschreibungssprache VHDL. Dazu lernen sie die Vor- und Nachteile von schematischer und textueller Hardware-Beschreibung kennen und können entscheiden, wann es sinnvoll ist, welche Variante einzusetzen. Darüber hinaus kombinieren sie synchrone und asynchrone Prozesse, um ein gutes Zusammenspiel der Komponenten ihres selbstgebauten Mikroprozessors zu erreichen. Schließlich beurteilen die Studierenden die Effizienz ihrer Implementierung anhand der erzielten Taktfrequenz und des benötigten Hardware-Aufwandes. In einer abschließenden Projektphase lernen sie, eine komplexe Aufgabenstellung zu planen, nach einem selbst entwickelten fundierten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zur verständlichen Darstellung und Dokumentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein, Akribie; Projektgebundene Arbeit und Zeitmanagement; Auswahl und sichere Anwendung geeigneter Methoden; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Selbstreflexion</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen: Modul Systemnahe Informatik (INF-0138) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile
<p>Modulteil: Hardware-Entwurf (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>
<p>Inhalte: Die Veranstaltung "Hardware-Entwurf" stellt Methoden des logischen Entwurfs digitaler Schaltungen dar, angefangen von der abstrakten Beschreibung in einer Hardwarebeschreibungssprache (wie VHDL) bis hin zur physikalischen Implementierung auf Transistorebene. Im praktischen Teil der Veranstaltung wird die Hardwareentwicklung am Beispiel einer fünfstufigen Prozessorpipeline veranschaulicht. Das Ergebnis ist ein in VHDL entwickelter lauffähiger Prozessor für ein FPGA-Prototypenboard</p>

Literatur:

- Uwe Brinkschulte, Theo Ungerer, Mikrocontroller und Mikroprozessoren, Springer Verlag, Heidelberg, dritte Auflage 2010
- John L. Hennessy, David A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 5. Auflage, 2011

Modulteil: Hardware-Entwurf (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Prüfung

Hardware-Entwurf

praktische Prüfung

Beschreibung:

Erfolgreiche Praktikumsteilnahme, Projektvorstellung am Ende des Semesters

Modul INF-0156: Algebraische Beschreibung paralleler Prozesse <i>Algebraic Description of Concurrent Processes</i>		6 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen:</p> <p>Die Studierenden kennen die Prozessalgebra CCS und formale Beschreibungen verteilter Systeme in CCS. Sie kennen einen Mechanismus, mit dem man in CCS die operationale Semantik definiert, und können diesen Mechanismus auf konkrete Prozesse anwenden. Sie besitzen die Fähigkeit, verteilte Systeme entsprechend auf eine exakte, algebraische Weise zu modellieren, d.h. einen entsprechenden Prozess zu entwerfen.</p> <p>Sie wissen, welche Anforderungen man in diesem Kontext an Äquivalenzbegriffe stellen muss und lernen verschiedene solche Begriffe kennen. Sie können auf verschiedenen Weisen -- z.B. auch durch algebraische Gesetze -- formal prüfen, ob ein System eine, ebenfalls in CCS geschriebene, Spezifikation erfüllt.</p> <p>Sie können beurteilen, ob ein eine algebraische Gleichung ein gültiges Gesetz ist. Zur Vertiefung der Kenntnisse werden die meisten Resultate der Vorlesung bewiesen.</p> <p>Die Studierenden wissen, wie man CCS und die operationale Semantik in einer spezifischen Weise um Zeit erweitern kann, und sind dadurch in der Lage, auch andere Prozessalgebren mit anderen Semantik-Regeln zu verstehen und zu verwenden.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Methodik; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit zur Analyse und Strukturierung von Informatikproblemstellungen; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 180 Std.</p> <p>15 Std. Übung (Präsenzstudium)</p> <p>45 Std. Vorlesung (Präsenzstudium)</p> <p>75 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>22 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>23 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
<p>Voraussetzungen:</p> <p>Modul Einführung in die Theoretische Informatik (INF-0110) - empfohlen</p> <p>Modul Logik für Informatiker (INF-0155) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 5.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Algebraische Beschreibung paralleler Prozesse (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Deutsch</p> <p>SWS: 3</p>		
<p>Inhalte:</p> <p>Algebraische Spezifikation verteilter Systeme mittels der Prozessalgebra CCS; operationale Semantik mittels SOS-Regeln; Äquivalenz- bzw. Kongruenzbegriffe (starke und schwache Bisimulation, Beobachtungskongruenz); Nachweis von Kongruenzen mittels Axiomen; Einführung in eine Kombination von Bisimulation und Effizienzvergleich.</p> <p>Die meisten Resultate der Vorlesung werden bewiesen.</p>		

Literatur:

- R. Milner: Communication and Concurrency, Prentice Hall
- L. Aceto, A. Ingolfsdottir, K.G. Larsen, J. Srba: Reactive Systems. Cambridge University Press 2007
- J. Bergstra, A. Ponse, S. Smolka (eds.): Handbook of Process Algebras, Elsevier

Modulteil: Algebraische Beschreibung paralleler Prozesse (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 1

Prüfung

Algebraische Beschreibung paralleler Prozesse (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0157: Endliche Automaten <i>Finite Automata</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen: Nach der Teilnahme haben die Studierenden vertiefte Kenntnisse in der Automatentheorie und können sie verständlich erläutern. Sie können deterministische Automaten minimieren und das Verfahren mit guter Effizienz automatisieren. Sie können Schaltkreisverhalten und Mealy-Automaten ineinander übersetzen, und sie können mit geeigneten Ergebnissen reguläre von nicht-regulären Sprachen unterscheiden. Zur Vertiefung der Kenntnisse werden die meisten Resultate der Vorlesung bewiesen. Sie haben anspruchsvolle Modellierungen von Problemen mit endlichen Automaten kennengelernt und können sich in neue Anwendungen der Automatentheorie einarbeiten und ihre Korrektheit beurteilen.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Methodik; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Auswahl und sichere Anwendung geeigneter Methoden; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 20 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 37 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 48 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 45 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen: Modul Einführung in die Theoretische Informatik (INF-0110) - empfohlen Modul Informatik 3 (INF-0111) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 5.	Minimale Dauer des Moduls: 1 Semester
SWS: 3	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Modulteil: Endliche Automaten (Vorlesung mit integrierter Übung) Lehrformen: Vorlesung + Übung Sprache: Deutsch SWS: 3</p>		
<p>Inhalte: Die Vorlesung vertieft die Kenntnisse über Endliche Automaten aus der Grundvorlesung "Einführung in die theoretische Informatik", insbesondere über Minimierung und Abschlusseigenschaften. Sie behandelt eine Anwendung bei der Lösung diophantischer Gleichungen. Sie stellt Mealy-, Moore- und Büchi-Automaten vor und behandelt die Übersetzungen zwischen Mealy-Automaten und Schaltkreisverhalten. Die meisten Resultate der Vorlesung werden bewiesen.</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • Hopcroft, (Motwani, Ullman: Introduction to Automata Theory, Languages and Computation; deutsch: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie • Schöning: Theoretische Informatik kurz gefaßt. 5. Auflage • Thomas: Automata on Infinite Objects. Chapter 4 in Handbook of Theoretical Computer Science, Hrsg. van Leeuwen 		
<p>Zugeordnete Lehrveranstaltungen: Endliche Automaten (Vorlesung + Übung)</p>		

Veranstaltung wird online/digital abgehalten.

Diese Vorlesung befasst sich mit endlichen Automaten in Ergänzung der Grundvorlesung "Einführung in die theoretische Informatik". Insbesondere werden die Minimierung endlicher Automaten, Automaten mit Ausgabe sowie Automaten für unendliche Wörter behandelt. Um die Nützlichkeit endlicher Automaten aufzuzeigen, wird z.B. kurz eine Verbindung zur Logik hergestellt.

Prüfung

Endliche Automaten (mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0161: Petrinetze - eine Theorie paralleler Systeme <i>Petri Nets - A Theory of Concurrent Systems</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen: Die Studierenden kennen verschiedene Varianten von Petrinetzen und entsprechende formale, Graphen-basierte Beschreibungen paralleler bzw. nebenläufiger Systeme. Anhand verschiedener Verhaltensbegriffe (Schalt- und Schrittfolgen, Sprache, Failure-Semantik) lernen sie die neuartigen Aspekte der Abläufe solcher Systeme kennen. Sie werden befähigt, wichtige Systemeigenschaften mit Petrinetz-spezifischen Methoden zu beweisen oder zu widerlegen. Auf dieser Grundlage können sie parallele bzw. nebenläufige Systeme mit Petrinetzen formal modellieren sowie analysieren, unter welchen Aspekten sich verschiedene Modelle gleichen oder unterscheiden. Sie lernen einige Entscheidbarkeitsprobleme kennen und können eigene Beweise dazu führen. Zur Vertiefung der Kenntnisse werden die meisten Resultate der Vorlesung bewiesen.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Methodik; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fertigkeit zur Analyse und Strukturierung von Informatikproblemstellungen; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Modul Einführung in die Theoretische Informatik (INF-0110) - empfohlen		
Angebotshäufigkeit: unregelmäßig (i. d. R. im WS)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Petrinetze - eine Theorie paralleler Systeme (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Graphenbasierte Modellierung paralleler Systeme mittels verschiedener Varianten von Petrinetzen; verschiedene Verhaltensbeschreibungen (Schalt- und Schrittfolgen, Sprache, Failure-Semantik); Begriffe und Techniken der Verhaltensanalyse (Verklemmung, Lebendigkeit, Fairness; S- und T-Invarianten, Überdeckbarkeitsgraph); Entscheidbarkeitsprobleme. Die meisten Resultate der Vorlesung werden bewiesen.</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • Desel, Reisig, Rozenberg (eds.): Lectures on Concurrency and Petri Nets. Advances in Petri Nets. Springer, LNCS 3098 • Peterson: Petri Net Theory and the Modelling of Systems. Prentice Hall • Reisig: Petrinetze - Eine Einführung. 2. Auflage; Springer 		

Modulteil: Petrinetze - eine Theorie paralleler Systeme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Petrinetze - eine Theorie paralleler Systeme (Mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0163: Verteilte Algorithmen <i>Distributed Algorithms</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen: Die Studierenden lernen formale Modelle zur Beschreibung verteilter Algorithmen in synchronen und asynchronen Netzwerke sowie die entsprechenden Methoden zum Nachweis der Korrektheit und des Aufwands kennen. Sie können beschreiben, wie Fehler in den beiden Modellen behandelt werden. Die Studierenden lernen wichtige Algorithmen und ihren Aufwand kennen, können sie in konkreten Netzwerken simulieren und verstehen ihre Korrektheit. Sie werden befähigt, solche Algorithmen zu modifizieren und zu analysieren, so dass sie zugehörige Korrektheitsbeweise und Aufwandsbestimmungen prüfen oder selbst entwickeln können. Sie erlangen damit ein vertieftes Verständnis für die Probleme und Problemlösungen in verteilten Systemen.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Methodik; Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fertigkeit zur Analyse und Strukturierung von Informatikproblemstellungen; Kenntnisse der Vor-/Nachteile von Entwurfsalternativen, Bewertung im jeweiligen Anwendungszusammenhang; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen: Modul Einführung in die Theoretische Informatik (INF-0110) - empfohlen Modul Informatik 3 (INF-0111) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Verteilte Algorithmen (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 4</p>		
<p>Inhalte: Algorithmen für Grundprobleme in Netzwerken wie Zugriff auf gemeinsame Ressourcen, Aufbau geeigneter Kommunikationsstrukturen und Konsens; es werden synchrone und asynchrone Netzwerke und Fehlertoleranz betrachtet, der Aufwand bestimmt und nachgewiesen sowie Korrektheitsbeweise geführt.</p>		
<p>Literatur: Nancy Lynch, Distributed Algorithms. Morgan Kaufmann 1996</p>		
<p>Zugeordnete Lehrveranstaltungen: Verteilte Algorithmen (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i> Diese Vorlesung präsentiert Algorithmen für Grundprobleme in Netzwerken mit zahlreichen Prozessoren, wie sie in den Bereichen Telekommunikation, Prozeßkontrolle und verteilte Informationssysteme auftreten, und untersucht</p>		

die Effizienz dieser Algorithmen. Grundprobleme sind der Zugriff auf gemeinsame Ressourcen (Problem des gegenseitigen Ausschlusses), Kommunikation (Aufbau geeigneter Strukturen zur Verbreitung von Daten im Netzwerk) und Konsens (z.B. ob eine Transaktion in einem verteilten Datenbanksystem abgebrochen werden muß). Dabei sollen Lösungen fehlertolerant sein, also auch bei fehlerhaftem Verhalten oder Ausfall einiger Prozessoren oder Kommunikationsverbindungen korrekt arbeiten. In der Vorlesung wird die Korrektheit der behandelten Algorithmen in der Regel bewiesen, da sie in vielen Anwendungen (in Telefon-, Reservierungs-, Bank- oder Luftfahrt-Kontroll-Systemen) essentiell ist.
... (weiter siehe Digicampus)

Modulteil: Verteilte Algorithmen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Verteilte Algorithmen (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Verteilte Algorithmen (Mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0179: Einführung in die Spieleprogrammierung <i>Foundations of Game Programming</i>		8 ECTS/LP
Version 1.1.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Studierenden die wesentlichen Konzepte zur Entwicklung eines Spiels. Sie sind in der Lage, eigene Gameplay-Konzepte umzusetzen und passende KI-Verfahren zu Entscheidungsfindung sowie Ablaufsteuerung einzusetzen. Sie verfügen über die Kenntnisse der Denkweise und die Sprache anwendungsrelevanter Disziplinen. Im Rahmen der Vorlesung erlernen sie, KI-Komponenten mit geeigneten Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmik selbstständig zu entwickeln und technisch umzusetzen. Besonders gefördert wird in diesem Rahmen auch die Fertigkeit zur sicheren und überzeugenden Darstellung von Ideen und Konzepten, verständlichen Darstellung und Dokumentation von Ergebnissen sowie zum logischen, analytischen und konzeptionellen Denken.</p> <p>Schlüsselqualifikationen: Auswahl und sichere Anwendung geeigneter Methoden, Fachübergreifende Kenntnisse, Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen, Fertigkeit zur Entwicklung und Umsetzung von Lösungsstrategien für komplexe Probleme, Verstehen von Teamprozessen, Fertigkeit der Zusammenarbeit in Teams, Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen, Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen: Ferienaufgabe		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Einführung in die Spieleprogrammierung (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
Inhalte: Game Engines, Entscheidungsfindung für KI-Charaktere, Wegfindung und Navigation, Gruppenverhalten und Gruppendynamik, Animationen, Physik, Storytelling, Ein-/Ausgabemethodik, Audio, Game Design, Grafik und Shaderprogrammierung		
Literatur: Skript		
Modulteil: Einführung in die Spieleprogrammierung (Übung)		
Lehrformen: Übung		
Sprache: Deutsch		
SWS: 4		

Prüfung

Einführung in die Spieleprogrammierung

Portfolioprüfung

Beschreibung:

Ausnahmefall SoSe 2020: Portfolioprüfung

Modul INF-0182: Praktikum Multimodale Echtzeitsignalverarbeitung <i>Practical Module Multimodal Real Time Signal Processing</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierenden mit den Grundkonzepten der Signalverarbeitung und des maschinellen Lernens vertraut. Sie sind in der Lage, fachliche Lösungskonzepte in Programme und Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Im Rahmen des Praktikums erlernen sie außerdem, in kleinen Teams größere Projektaufgaben (Entwicklung von Softwaremodulen) zu planen, nach einem selbst entwickelten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und als Team zu präsentieren. Besonders gefördert wird zudem die Fähigkeit geeignete Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmik selbstständig zu entwickeln und technisch umzusetzen.		
Schlüsselqualifikationen: Fortgeschrittene Kenntnisse der Signalverarbeitung und des maschinellen Lernens, Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen, Fertigkeit zur Entwicklung und Umsetzung von Lösungsstrategien für komplexe Probleme, Verstehen von Teamprozessen, Fertigkeit der Zusammenarbeit in Teams, Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen, Qualitätsbewusstsein, Akribie		
Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Praktikum (Präsenzstudium) 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)		
Voraussetzungen: Programmiererfahrung Modul Multimedia Grundlagen I (INF-0087) - empfohlen Modul Multimedia Grundlagen II (INF-0166) - empfohlen		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Multimodale Echtzeitsignalverarbeitung Lehrformen: Praktikum Sprache: Deutsch SWS: 6		
Inhalte: Die konkrete Aufgabenstellung aus dem weiten Gebiet der multimodalen Echtzeitsignalverarbeitung wird jedes Jahr neu entworfen.		
Literatur: Literaturhinweise werden je nach Thema zu Beginn des Semesters bekanntgegeben.		
Prüfung Praktikum Multimodale Echtzeitsignalverarbeitung praktische Prüfung		

Modul INF-0183: Praktikum Spieleprogrammierung <i>Practical Module Game Programming</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Studierenden die wesentlichen Konzepte der Spieleentwicklung. Sie sind in der Lage, in kleinen Teams größere Projektaufgaben zu planen, nach einem selbst entwickelten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren. Sie sind in der Lage, fachliche Lösungskonzepte in Programme und Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Sie verfügen über die Kenntnisse der Denkweise und die Sprache anwendungsrelevanter Disziplinen. Im Rahmen des Praktikums erlernen sie, verschiedenste Komponenten eines Spiels mit geeigneten Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmik selbstständig zu entwickeln und technisch umzusetzen. Besonders gefördert wird in diesem Rahmen auch die Fertigkeit zur sicheren und überzeugenden Darstellung von Ideen und Konzepten, verständlichen Darstellung und Dokumentation von Ergebnissen sowie zum logischen, analytischen und konzeptionellen Denken.</p> <p>Schlüsselqualifikationen: Fertigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten; Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen; Verstehen von Teamprozessen; Fertigkeit der Zusammenarbeit in Teams; Fähigkeit zur Leitung von Teams; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Fähigkeit, Beiträge zur Wissenschaft zu leisten; Kompetenz zum Erkennen von bedeutendetechnischen Entwicklungen; Qualitätsbewusstsein, Akribie.</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Praktikum (Präsenzstudium) 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: Modul Einführung in die Spieleprogrammierung (INF-0179) - empfohlen		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 2.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Praktikum Spieleprogrammierung Lehrformen: Praktikum Sprache: Deutsch SWS: 6</p>		
Inhalte: Innerhalb des Praktikums soll ein Spiel entwickelt werden. Der inhaltliche Schwerpunkt des Praktikums wird jedes Jahr neu festgelegt.		
Literatur: Literaturhinweise werden zu Beginn des Semesters bekanntgegeben.		
<p>Zugeordnete Lehrveranstaltungen: Praktikum Spieleprogrammierung (Praktikum) <i>*Veranstaltung wird in Präsenz abgehalten.*</i> Informationen zur Veranstaltung finden Sie auf unserer Lehrstuhlwebsite: https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/hcm/studies/lehrveranstaltungen/praktikum-spieleprogrammierung/</p>		

Prüfung

Praktikum Spieleprogrammierung

praktische Prüfung

Modul INF-0189: Qualitätssicherung im Software Engineering <i>Quality Assurance in Software Engineering</i>		5 ECTS/LP
Version 2.0.0 (seit SoSe18) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Nach erfolgreicher Absolvierung der Lehrveranstaltung verstehen Studierende die Grundtechniken der Qualitätssicherung und der Problemstellungen des Testens in der Praxis. Sie sind in der Lage, SW-Module zu spezifizieren und kennen die wesentlichen Testtechniken und deren Einsatzzwecke im Software Engineering. Sie können Testkonzepte für große Softwaresysteme entwerfen, modellieren und anwenden. Die Studierenden sind für das Thema Qualität im Software Engineering sensibilisiert und können verschiedene Qualitätskriterien/-metriken kritisch hinterfragen und bewerten.</p> <p>Des Weiteren kennen und verstehen sie die Prinzipien von konstruktiven Qualitätssicherungstechniken und -praktiken.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Training des logischen Denkens • Analytisch-methodische Kompetenz • Abwägen von Vorgehensweisen • Erwerb von Abstraktionsfähigkeiten • Verbesserung der eigenen Softwareentwicklungskompetenz 		
<p>Arbeitsaufwand: Gesamt: 150 Std. 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Modulteil: Qualitätssicherung im Software Engineering (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		

Inhalte:

Ingenieursdisziplinen kombinieren Design- und Entwicklungsaktivitäten mit Aktivitäten, die vorläufige und endgültige Produkte prüfen, um Mängel zu erkennen und zu beseitigen. Software Engineering ist hierbei keine Ausnahme: Konstruktion hochqualitativer Software bedarf einer sich ergänzenden Kombination von Maßnahmen des Designs und der Prüfung der Software über den gesamten Entwicklungszyklus hinweg. Gerade aufgrund der Durchdringung der Software von immer mehr kritischen Bereichen wie etwa Automotive oder Avionik rücken Maßnahmen zu Qualitätssicherung immer mehr in den Fokus der Aufmerksamkeit im modernen Software Engineering. In dieser Vorlesung werden Konzepte, Techniken und Methoden der Qualitätssicherung im Software Engineering vermittelt. Dies umfasst, u.a., die Spezifikation von Software in einem Kontinuum von natürlichsprachlicher bis formalsprachlicher Notation, automatisierte Methoden und Techniken zur analytischen sowie auch zur konstruktivistischen Qualitätssicherung, Entwicklung von Qualitätssicherungsstrategien sowie Grundlagen im Umgang mit gängigen Werkzeugen, die im Software Engineering zum Einsatz kommen. Den Abschluss bildet die kritische Auseinandersetzung mit formalen Methoden, die für besonders kritische Module zum Einsatz kommen können und in Zertifizierungsstandards anerkannt werden.

Literatur:

- P. Ammann und J. Offutt: Introduction to Software Testing. Cambridge University Press, 2008.
- M. Pezzè und M. Young: Software Testing and Analysis: Process, Principles, and Techniques. Wiley & Sons, 2008.
- R. Binder: Testing Object-Oriented Systems: Models, Patterns, and Tools. Addison-Wesley, 2000.
- M. Chemuturi: Mastering Software Quality Assurance: Best Practices, Tools and Techniques for Software Developers. J. Ross Publishing, 2011.
- G. O'Regan: Introduction to Software Quality. Springer, 2014.
- W. Reif: Software-Verifikation und ihre Anwendungen, it+ti Themenheft, Oldenbourg Verlag, 1997
- Vorlesungsskript
- In der Vorlesung bereitgestellte wiss. Publikationen, Journalartikel und Buchbeiträge.

Modulteil: Qualitätssicherung im Software Engineering (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Qualitätssicherung im Software Engineering (Mündliche Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0201: Platzeffiziente Algorithmen <i>Space-Efficient Algorithms</i>		5 ECTS/LP
Version 1.1.0 (seit WS15/16) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
<p>Lernziele/Kompetenzen: Die Studierenden sind in der Lage, platzeffiziente Algorithmen zu verstehen, zu analysieren und selbst zu entwerfen. Sie verstehen die häufig notwendige Abwägung zwischen Zeit und Platz und kennen wichtige Entwurfsmethoden und grundlegende Datenstrukturen für platzeffiziente Algorithmen ebenso wie eine Anzahl konkreter platzeffizienter Algorithmen.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
<p>Voraussetzungen: Teilnehmer der Vorlesung sollten über gute algorithmische Kenntnisse verfügen, insbesondere im Bereich Graphenalgorithmen. Modul Informatik 3 (INF-0111) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Platzeffiziente Algorithmen (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Manchmal hat ein Algorithmus eine große Eingabe, aber nur wenig frei beschreibbaren Arbeitsspeicher. Zum Beispiel könnte die Eingabe im Internet für Anfragen zur Verfügung stehen, aber in ihrer Gesamtheit so riesig sein, dass es unmöglich oder unpraktisch ist, sie auf den lokalen Rechner herunterzuladen. Die Vorlesung beschäftigt sich aus theoretischer Sicht mit Algorithmen, die mit weniger Arbeitsspeicher als klassische Algorithmen für dieselben Probleme auskommen. Der Fokus liegt auf Graphenproblemen wie die Durchführung einer Tiefensuche oder die Berechnung kürzester Wege, aber auch Sortieren und platzeffiziente Datenstrukturen kommen zur Sprache. Ein Großteil der in der Vorlesung vorgestellten Ergebnisse wurde seit 2014 am Lehrstuhl für Theoretische Informatik erzielt. Die Vorlesung behandelt somit ein sehr aktives und aktuelles Forschungsgebiet.</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • Skript 		

Modulteil: Platzeffiziente Algorithmen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Platzeffiziente Algorithmen (mündliche Prüfung)

Mündliche Prüfung, Dauer: 30-45 Minuten

Beschreibung:

Die Prüfung findet zum Ende des Semesters statt. Es besteht bei Bedarf die Möglichkeit einer Nachholprüfung im Folgesemester.

Modul INF-0207: Reinforcement Learning <i>Reinforcement Learning</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Studierenden die wesentlichen Konzepte des Reinforcement Learnings. Sie sind in der Lage, fachliche Lösungskonzepte in Programme und Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Sie verfügen über die Kenntnisse der Denkweise und die Sprache anwendungsrelevanter Disziplinen. Im Rahmen der Vorlesung erlernen sie, lernende Komponenten mit geeigneten Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmen selbstständig zu entwickeln und technisch umzusetzen. Besonders gefördert wird in diesem Rahmen auch die Fertigkeit zur sicheren und überzeugenden Darstellung von Ideen und Konzepten, verständlichen Darstellung und Dokumentation von Ergebnissen sowie zum logischen, analytischen und konzeptionellen Denken.</p> <p>Schlüsselqualifikationen: Fortgeschrittene mathematisch-formale Methodik, Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen, Fertigkeit zur Entwicklung und Umsetzung von Lösungsstrategien für komplexe Probleme, Verstehen von Teamprozessen, Fertigkeit der Zusammenarbeit in Teams, Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen, Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 60 Std. Übung (Präsenzstudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Reinforcement Learning (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
Inhalte: Markov-Entscheidungsproblem, Dynamische Programmierung, Monte Carlo Methoden, TD-Lernen, Eligibility Traces, Hierarchisches Bestärkendes Lernen, Planen und Lernen, Generalisierung		
Literatur: <ul style="list-style-type: none"> • Skript • Richard S. Sutton und Andrew G. Barto. Reinforcement Learning: An Introduction. MIT Press, 1998 		
Zugeordnete Lehrveranstaltungen:		
Reinforcement Learning (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i>		

Inhalt der Veranstaltung sind Methoden und Prinzipien des Reinforcement Learning. In der Übung werden verschiedene Probleme modelliert, mit Hilfe von Lernalgorithmen als Simulationen implementiert und verglichen. Siehe <https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/hcm/studies/lehveranstaltungen/reinforcement-learning/>

Modulteil: Reinforcement Learning (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Zugeordnete Lehrveranstaltungen:

Übung zu Reinforcement Learning (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Reinforcement Learning

Portfolioprüfung

Modul INF-0213: Parallele und Verteilte Datenbanksysteme <i>Parallel and Distributed Database Systems</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe16) Modulverantwortliche/r: PD Dr. Markus Endres		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an der Veranstaltung sind die Studierenden in der Lage die Konzepte und Methoden, Verfahren, Techniken und Technologien von parallelen und verteilten Datenbanksystemen im Detail zu verstehen, zu bewerten und anzuwenden. Darüber hinaus können die Studierenden weiterführende komplexe Problemstellungen auf dem Gebiet Datenbanken, insbesondere unter Verwendung von parallelen und verteilten Algorithmen, analysieren und lösen. Die Vorlesung vermittelt zudem die Fertigkeit zum logischen, analytischen und konzeptionellen Denken. Bei den in der Veranstaltung anfallenden praxisnahen Programmieraufgaben ist eine sichere und überzeugende Darstellung von Ideen und Konzepten sowie die Zusammenarbeit in Teams entscheidend.</p> <p>Schlüsselqualifikationen: Fachspezifische Vertiefung; Kenntnisse der Vor-/Nachteile von Entwurfalternativen und deren Bewertung im jeweiligen Anwendungszusammenhang; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen; Fähigkeit, vorhandenes Wissen selbständig zu erweitern;</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 3.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Parallele und Verteilte Datenbanksysteme (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Vorlesung behandelt parallele und verteilte Datenbanksysteme. Dazu wird die Architektur von parallelen und verteilten Datenbanksystemen vorgestellt. Der Schwerpunkt liegt anschließend bei der Ausarbeitung und Implementierung von parallelen und verteilten Algorithmen. Insbesondere werden hierbei die parallele Suche, parallele Gruppierung, parallele Joins, Transaktionen in verteilten Datenbanken, verteilte Transaktionsprotokolle, paralleles OLAP, paralleles Data Mining und paralleles Clustering und Klassifikation behandelt.</p>		

Literatur:

- - Taniar et al.: High-Performance Parallel Database Processing and Grid Databases
- - E. Rahm: Mehrrechner-Datenbanksysteme
- - M. T. Özsu, P. Valduriez: Principles of Distributed Database Systems
- - P. Dada: Verteilte Datenbanken und Client/Server-Systeme
- - S. Conrad: Förderierte Datenbanksysteme
- - S. K. Rahimi, F. S. Haug: Distributed Database Management Systems - A Practical Approach

Modulteil: Parallele und Verteilte Datenbanksysteme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Parallele und Verteilte Datenbanksysteme (mdl. Prüfung)

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0216: Vertiefte Multicore-Programmierung <i>Advanced Multicore Programming</i>		8 ECTS/LP
Version 2.0.0 (seit SoSe20 bis WS20/21) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen: Die Studierenden erwerben Kompetenzen auf einem fortgeschrittenen, wissenschaftlichen Niveau in folgenden Bereichen: Parallelprogrammierung von speichergekoppelten Systemen in C++11 und OpenMP, Entwurf von lock-freien Algorithmen, Programmierung eines Hardware-Transaktionsspeicher-Systems am Beispiel von Intel TSX, nachrichtengekoppelte Programmierung mit dem Message-Passing-Interface (MPI), Programmierung von GPU-Beschleunigerkarten mit Nvidia CUDA.</p> <p>Sie können die Vor- und Nachteile der unterschiedlichen Programmiermodelle und Architekturen einschätzen, und parallele Programme in den jeweiligen Sprachen analysieren. Durch die praktischen Übungen setzen die Studierenden die erlernten Konzepte direkt in die Praxis um. Dabei erwerben die Studierenden fortgeschrittene Programmierkenntnisse in C++11, Intel TSX, MPI, OpenMP und CUDA. Sie können selbständig das Laufzeitverhalten der jeweiligen Programmiermodelle bewerten und optimieren. In einer abschließenden Projektphase lernen die Studierenden, eine komplexe Aufgabenstellung im Team zu planen, nach einem selbst entwickelten Lösungsansatz zu bearbeiten und die Resultate in wissenschaftlicher Weise im Plenum zu diskutieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Analytische-methodische Kompetenz im Bereich der Multicore-Programmierung von CPUs und GPUs; Projektgebundene Arbeit und Zeitmanagement; Auswahl und sichere Anwendung geeigneter Methoden; Fähigkeit, vorhandenes Wissen selbständig zu kombinieren und zu erweitern; Selbstreflexion</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 60 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Vertiefte Multicore-Programmierung (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Vorlesung vertieft verschiedene Techniken der Parallelprogrammierung für aktuelle Multicore-Prozessoren und Grafikkarten. Nach einer grundlegenden Einführung in Threads, Synchronisationskonstrukte und weiterführende Konzepte der Parallelprogrammierung in C++11 werden weitere parallele Programmiermodelle behandelt.</p>		

Literatur:

- M. Herlihy, N. Shavit: The Art of Multiprocessor-Programming, Morgan Kaufmann, 2012, ISBN: 978-0123973375
- M. McCool, J. Reinders, A. D. Robison: Structured Parallel Programming: Patterns for Efficient Computation, Morgan Kaufmann, 2012, ISBN: 978-0124159938
- T. Rauber, G. Runger: Parallele Programmierung, Springer Verlag, 2007, ISBN: 978-3540465492
- es werden die jeweils neuesten APIs/Unterlagen aus dem Internet verwendet

Modulteil: Vertiefte Multicore-Programmierung (bung)

Lehrformen: bung

Sprache: Deutsch

SWS: 4

Inhalte:

Ziel der bung ist es, den Umgang mit den unterschiedlichen Programmiermodellen sowie Performanzanalyse- und Debugging-Techniken in praktischen Beispielen zu vertiefen. Die bung wird durch eine Projektphase abgeschlossen, die es den Studierenden ermoglicht, die behandelten Programmier Techniken in einem umfangreicheren Projekt selbstandig anzuwenden, Ergebnisse auszuwerten und zu prasentieren.

Prufung

Vertiefte Multicore-Programmierung

Portfolioprufung

Beschreibung:

Projektvorstellung und -abnahme, Fragen zu Vorlesung und bung

Modul INF-0217: Praktikum Autonomes Fahren <i>Practical Module Autonomous Driving</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe16) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Praktikum Autonomes Fahren verstehen es die Studierenden praxisnahe Problemstellungen hoher Komplexität im Bereich der Konzeptionierung, Entwicklung und Absicherung von hochautomatisierten/ autonomen Fahrzeugen mit aktuellen Methoden und Tools der modellbasierten Entwicklung zu lösen. Die Studierenden erlangen tiefgehende fachspezifische als auch fächerübergreifende Kenntnisse und Fähigkeiten, beispielsweise aus der Hardwarenahen Informatik, dem Software Engineering, als auch der zugrundeliegenden Fahrphysik und Mathematik. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren, zu präsentieren und verständlich zu dokumentieren.</p> <p>Schlüsselqualifikation: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Projektmanagementfähigkeiten</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 150 Std. Praktikum (Präsenzstudium)</p>		
<p>Voraussetzungen: Empfohlen wird die Teilnahme an einem der beiden Seminare. Modul Seminar Grundlagen des Software Engineering für Automotive Systems (BA) (INF-0027) - empfohlen Modul Seminar Grundlagen des Software Engineering für Automotive Systems (MA) (INF-0040) - empfohlen</p>		
<p>Angebotshäufigkeit: jedes Sommersemester</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 10</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p> <p>Modulteil: Praktikum Autonomes Fahren Lehrformen: Praktikum Sprache: Deutsch / Englisch SWS: 10</p>		

Inhalte:

In diesem Praktikum lernen die Teilnehmer, wie verschiedene ausgewählte Teilaspekte des Autonomen Fahrens umgesetzt, simuliert und analysiert werden können.

In einem Einführungskurs werden die benötigten theoretischen Grundlagen für die Bearbeitung der Praxisaufgaben gelegt.

Darüber hinaus lernen die Teilnehmer u.a. das im Automotive-Umfeld häufig eingesetzte Entwicklungswerkzeug „ADTF“ von Elektrobit und die darin verwendete Programmiersprache C++ kennen.

Nach dem Einführungskurs sollen in Gruppen von zwei bis drei Teilnehmern mithilfe der genannten Werkzeuge autonome Fahrfunktionen umgesetzt werden.

Die entwickelten Ergebnisse werden abschließend an 1:8 Fahrzeugmodellen, deren Sensorik realen Fahrzeugen sehr nahe kommt, demonstriert und ausgewertet.

Prüfung

Praktikum Autonomes Fahren

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0233: Industrierobotik <i>Industrial Robotics</i>		8 ECTS/LP
Version 2.2.0 (seit SoSe17) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
Lernziele/Kompetenzen: Die Studierenden sind in der Lage Industrieroboter zu programmieren, kennen das dafür erforderliche methodische Vorgehen und können die Eignung verschiedener Vorgehensweise bewerten. Sie können fachliche Lösungskonzepte in Roboterprogramme umsetzen und dabei Entwurfsalternativen bewerten und anwenden. Sie haben Kenntnisse von praxisrelevanten Aufgabenstellungen in der automatisierten Fertigung. Sie haben die Fertigkeit zum analytischen und konzeptionellen Denken.		
Schlüsselqualifikationen: <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Erwerb von Abstraktionsfähigkeiten 		
Bemerkung: Dieses Modul ersetzt das Modul "INF-0132: Software in Mechatronik und Robotik". Sofern dieses bereits belegt wurde, ist eine erneute Belegung nicht möglich.		
Arbeitsaufwand: Gesamt: 240 Std. 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Übung (Präsenzstudium)		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Industrierobotik (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
Inhalte: Ziel der Veranstaltung ist es, an Beispielen die Programmierung und der Entwurf von Software für Industrieroboter, wie sie z.B. in der Automobilindustrie verwendet werden, zu erlernen. Dazu werden im ersten Teil der Vorlesung Grundlagen wie Kinematik und Bahnplanung mit Hilfe simulationsbasierter Programmieransätze behandelt. Im zweiten Teil der Vorlesung werden verschiedene, kleine Programmieraufgabenstellungen bearbeitet und auf einem KUKA Roboter evaluiert. Die Programmierung erfolgt mit der Roboterprogrammiersprache KRL.		
Literatur: <ul style="list-style-type: none"> • L. Sciavicco, B. Siciliano: Modelling and Control of Robot Manipulators. Reihe: Advanced Textbooks in Control and Signal Processing. Springer 2000 (2nd Ed.) • Handbücher von KUKA • Folienhandout 		

Zugeordnete Lehrveranstaltungen:

Industrierobotik (Vorlesung)

Veranstaltung wird als Hybrid/gemischt abgehalten.

Diese Veranstaltung steht unter dem Motto "Uni goes Industry", da in dieser Veranstaltung die Programmierung "echter" Industrieroboter incl. zugehöriger Software-Entwicklungsumgebung und Simulationsmöglichkeiten vermittelt wird. Dazu werden in Zweiergruppen verschiedene, kleine Programmieraufgabenstellungen bearbeitet und auf einem KUKA KR 6 Roboter evaluiert. Zudem werden wichtige Grundlagen der Robotik wie Kinematik und Bahnplanung anhand eines simulierten Roboters behandelt. Ziel der Veranstaltung ist es, an Beispielen die Programmierung und der Entwurf von Software für Industrieroboter, wie sie z.B. in der Automobilindustrie verwendet werden, zu erlernen. Die Vorlesung "Industrierobotik" ersetzt die Veranstaltung "Software in Mechatronik und Robotik". Studierende, die an der Vorlesung "Software in Mechatronik und Robotik" teilgenommen haben, können diese Veranstaltung nicht mehr besuchen.

... (weiter siehe Digicampus)

Modulteil: Industrierobotik (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Zugeordnete Lehrveranstaltungen:

Übung zu Industrierobotik (Übung)

Veranstaltung wird als Hybrid/gemischt abgehalten.

Prüfung

Industrierobotik

praktische Prüfung / Prüfungsdauer: 45 Minuten

Bearbeitungsfrist: 3 Monate

Beschreibung:

Ausnahmefall WiSe 2020/2021: Die Prüfung findet als Portfolioprüfung statt.

Modul INF-0235: Software für Industrie 4.0 <i>Software for Industry 4.0</i>		5 ECTS/LP
Version 1.2.0 (seit WS16/17) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Die Studierenden sind in der Lage die praktischen und methodischen Herausforderungen, die Industrie 4.0 für die Automatisierung und deren Softwareentwicklung bedeutet, zu verstehen. Sie werden sowohl in Techniken der klassischen Automatisierung als auch modernen Softwaretechnologien im Umfeld von Industrie 4.0 eingeführt und kennen so unterschiedliche anwendungsrelevante Disziplinen. Sie sind in der Lage Lösungskonzepte zu erstellen und dabei Entwurfsalternativen bewerten und anwenden.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenzen • Abwägen von Lösungsansätzen • Erwerb von Abstraktionsfähigkeiten 		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Software für Industrie 4.0 (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Vorlesung behandelt aktuelle Themen im Kontext von Industrie 4.0. Dazu zählen folgende Inhalte:</p> <ul style="list-style-type: none"> • Einführung in die Automatisierung • Referenzarchitekturen für Industrie 4.0 • Einführung in die (mobile Service) Robotik • OPC UA • AutomationML • Data Analytics für Industrie 4.0 		
<p>Literatur:</p> <ul style="list-style-type: none"> • Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0 - Abschlussbericht des Arbeitskreises Industrie 4.0; acaTech 		

Modulteil: Software für Industrie 4.0 (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Software für Industrie 4.0

Mündliche Prüfung / Prüfungsdauer: 45 Minuten

Modul INF-0243: Process Mining <i>Process Mining</i>		6 ECTS/LP
Version 1.1.0 (seit SoSe17) Modulverantwortliche/r: Prof. Dr. Robert Lorenz		
<p>Lernziele/Kompetenzen: Teilnehmer verstehen die folgenden wesentlichen Konzepte der Informatik auf einem vertieften wissenschaftlichen Niveau mit ihren mathematisch-formalen Grundlagen: Ereignis-Log, Halbordnung und partielle Sprache, Petrinetz, Nebenläufigkeit, sequentielle und kausale Semantik eines nebenläufigen Systems, Synthese von nebenläufigen Systemen, Geschäftsprozess, Process Mining, Process Discovery. Sie können nebenläufige Systeme mittlerer Komplexität in einer geeigneten Modellierungssprache modellieren, sowie zu einem gegebenen nebenläufigen System verschiedene Verhaltensmodelle zur Analyse und Simulation generieren, bewerten und ineinander überführen. Sie kennen die verschiedenen Process Mining Anwendungsfälle mit dazu passenden Lösungstechniken und Qualitätskriterien zu deren Bewertung. Teilnehmer verfügen über eine fortgeschrittene mathematisch formale Methodik für die Analyse und Formalisierung komplexer Process Discovery Probleme. Sie sind in der Lage, geeignete Algorithmen zu deren Lösung bzgl. qualitativer und quantitativer Kriterien sicher begründet auszuwählen, sowie diese anzuwenden und die Ergebnisse zu interpretieren und zu bewerten.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen, konzeptionellen und algorithmischen Denken; Eigenständiges Arbeiten mit Lehrbüchern und englischsprachiger Fachliteratur; Verständliche Präsentation von Ergebnissen; Qualitätsbewußtsein; Akribie; Grundsätze guter wissenschaftlicher Praxis; Wissenschaftliche Methodik;</p>		
<p>Arbeitsaufwand: Gesamt: 180 Std. 45 Std. Vorlesung (Präsenzstudium) 15 Std. Übung (Präsenzstudium) 23 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 75 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 22 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile
<p>Modulteil: Process Mining (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch / Englisch SWS: 3</p>
<p>Inhalte: Die Veranstaltung gibt einen fundierten Überblick über das Forschungsgebiet Process Mining:</p> <ul style="list-style-type: none"> - Modelling Techniques: Workflow Nets, Partial Languages - Discovering algorithms - Conformance Checking - Process Enhancement - The PROM Framework

Literatur:

Wil M. P. van der Aalst:

Process Mining - Data Science in Action, Second Edition. Springer 2016, ISBN 978-3-662-49850-7

Wil M. P. van der Aalst, Boudewijn F. van Dongen: Discovering Petri Nets from Event Logs. Trans. Petri Nets and Other Models of Concurrency 7: 372-422 (2013)

Robin Bergenthum, Jörg Desel, Robert Lorenz, Sebastian Mauser: Process Mining Based on Regions of Languages. BPM 2007: 375-383

Modulteil: Process Mining (Übung)

Lehrformen: Übung

Sprache: Deutsch / Englisch

SWS: 1

Prüfung

Process Mining (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0248: Kollaborative Robotik <i>Collaborative Robotics</i>		8 ECTS/LP
Version 1.1.0 (seit SoSe17) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Die Studierenden sind in der Lage kollaborative Roboter zu programmieren und verstehen die regulatorischen Randbedingungen (vgl. ISO/TS 15066:2016). Sie verstehen die grundlegenden Konzepte und Methoden der kollaborativen Robotik. Sie können dementsprechend fachliche Lösungskonzepte umsetzen und Entwurfsalternativen bewerten und anwenden. Sie haben die Fertigkeit zum analytischen und konzeptionellen Denken und können grundlegende Bewertungen der Risiken eines kollaborativen Robotersystems durchführen. Die Studierenden können abwägen, für welche praxisrelevanten Problemstellungen kollaborative Roboter eingesetzt werden können.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Erwerb von Abstraktionsfähigkeiten • Erwerb von Grundlagen für die Sicherheitsbewertung 		
<p>Bemerkung: Voraussetzungen (empfohlen): Software für Industrie 4.0 oder Industrierobotik oder Grundlagen der Robotik</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std.</p>		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Kollaborative Robotik (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 4</p>		

Inhalte:

Die Digitalisierung dringt mittlerweile in viele Bereiche des täglichen Lebens vor. Von Industriemaschinen und Robotersystemen (*Smart Factory, Industrie 4.0*) bis hin zu intelligenten Hausgeräten oder Heizungsanlagen (*Smart Home*) findet eine immer stärkere Vernetzung dieser Geräte im Internet der Dinge statt. Hierbei spricht man von cyber-physischen Systemen, die einerseits durch ein komplexes Zusammenspiel von vernetzten eingebetteten Systemen entstehen und andererseits geprägt sein werden von einer vollkommen neuen Art der Mensch-Technik-Interaktion in den Anwendungen.

Eine besondere Rolle unter den cyber-physischen Systemen nehmen Assistenz- oder Serviceroboter ein, da sie durch einen geteilten Arbeitsraum in eine direkte, auch physische Interaktion mit dem Menschen treten können. Diese robotischen Systeme können den Menschen in seiner täglichen Arbeit unterstützen bzw. entlasten. Das gilt sowohl im häuslichen Umfeld als auch in der Fabrik. Im Kontext von Industrie 4.0 haben robotische Co-Worker (CoBots) vereinzelt bereits Eingang in die Produktion gefunden. Diese kollaborativen Einsatzszenarien werden jedoch in Zukunft noch weiter an Bedeutung gewinnen. Durch die in 2016 veröffentlichte Norm ISO/TS 15066:2016 zur Mensch-Roboter-Kollaboration stehen dem Einsatz innovativer Robotersysteme neue Möglichkeiten offen.

Während die grundlegende Hardware, d.h. die Roboter, mobilen Plattformen, Greifer und Sensoren, bereitgestellt wird, liegt der Schwerpunkt der Veranstaltung auf der Entwicklung von Software für kollaborative (mobile) Manipulatoren. Durch die intelligente Vernetzung von Aktuatorik und Sensorik, neuartige Algorithmen für die Sensorauswertung und kreative Konzepte zur (physischen) Interaktion mit dem Menschen entstehen innovative, kollaborative Robotersysteme im praktischen Teil der Veranstaltung.

Infolgedessen liegt der thematische Fokus der Veranstaltung auf

- der Umsetzung multi-modaler Benutzerschnittstellen,
- der Gestaltung neuartiger, auch physischer Interaktionsmöglichkeiten,
- der Realisierung intelligenter Verhaltensweisen durch Techniken der Künstlichen Intelligenz bzw. des Machine Learnings und
- der Entwicklung kognitiver Fähigkeiten des Robotersystems.

Diese Kernelemente bieten die Grundlage, um eine praktische Aufgabenstellung prototypisch umzusetzen. Die Programmierung erfolgt dabei in objektorientierten Programmiersprachen. So können kollaborative Roboter bspw. in Java mit der Robotics API programmiert werden, einem am ISSE in Kooperation mit KUKA entwickelten Roboterframework. Alternativ oder ergänzend kann das Robot Operating System (ROS) verwendet werden, das in den Programmiersprachen C++ und Python programmiert werden kann.

Zugeordnete Lehrveranstaltungen:**Übung zu Kollaborative Robotik (Übung)**

Veranstaltung wird als Hybrid/gemischt abgehalten.

Modulteil: Kollaborative Robotik (Vorlesung)

Lehrformen: Vorlesung

Sprache: Deutsch

SWS: 2

Literatur:

- ISO/TS 15066:2016. 2016-02-15. Robots and robotic devices - Collaborative robots
- DIN EN ISO 13849-1:2016-06. 06.2016. Sicherheit von Maschinen - Sicherheitsbezogene Teile von Steuerungen - Teil 1: Allgemeine Gestaltungsleitsätze (ISO 13849-1:2015); Deutsche Fassung EN ISO 13849-1:2015
- DIN EN ISO 10218-1:2012. 01.2012. Industrieroboter - Sicherheitsanforderungen - Teil 1: Roboter (ISO 10218-1:2011); Deutsche Fassung EN ISO 10218-1:2011

Zugeordnete Lehrveranstaltungen:**Kollaborative Robotik (Vorlesung)**

Veranstaltung wird als Hybrid/gemischt abgehalten.

Der Schwerpunkt der Veranstaltung liegt auf der Entwicklung von Software für kollaborative (mobile) Manipulatoren; die grundlegende Hardware, d.h. die Roboter, mobilen Plattformen, Greifer und Sensoren, wird bereitgestellt. Durch die intelligente Vernetzung von Aktuatorik und Sensorik, neuartige Algorithmen für die Sensorauswertung und kreative Konzepte zur (physischen) Interaktion mit dem Menschen entstehen innovative, kollaborative Robotersysteme im praktischen Teil der Veranstaltung. Infolgedessen liegt der thematische Fokus der Veranstaltung auf • der Umsetzung multi-modaler Benutzerschnittstellen, • der Gestaltung neuartiger, auch physischer Interaktionsmöglichkeiten, • der Realisierung intelligenter Verhaltensweisen durch Techniken der Künstlichen Intelligenz bzw. des Machine Learnings und • der Entwicklung kognitiver Fähigkeiten des Robotersystems. Diese Kernelemente bieten die Grundlage, um eine praktische Aufgabenstellung prototypisch umzusetzen. Die Programmierung erfolgt dab
... (weiter siehe Digicampus)

Prüfung

Kollaborative Robotik

praktische Prüfung / Prüfungsdauer: 60 Minuten

Bearbeitungsfrist: 3 Monate

Modul INF-0250: Praktikum Reinforcement Learning <i>Practical Module Reinforcement Learning</i>		8 ECTS/LP
Version 1.3.0 (seit SoSe17) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Studierenden die wesentlichen Konzepte des Reinforcement Learnings. Sie sind in der Lage, in kleinen Teams größere Projektaufgaben zu planen, nach einem selbst entwickelten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren. Sie sind in der Lage, fachliche Lösungskonzepte in Programme und Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Sie verfügen über die Kenntnisse der Denkweise und die Sprache anwendungsrelevanter Disziplinen. Im Rahmen des Praktikums erlernen sie, lernende Komponenten mit geeigneten Methoden wissenschaftlich aussagekräftig zu bewerten, die Methoden und Algorithmen selbstständig zu entwickeln und technisch umzusetzen. Besonders gefördert wird in diesem Rahmen auch die Fertigkeit zur sicheren und überzeugenden Darstellung von Ideen und Konzepten, verständlichen Darstellung und Dokumentation von Ergebnissen sowie zum logischen, analytischen und konzeptionellen Denken.</p> <p>Schlüsselqualifikationen: Fertigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten; Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen; Verstehen von Teamprozessen; Fertigkeit der Zusammenarbeit in Teams; Fähigkeit zur Leitung von Teams; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Fähigkeit, Beiträge zur Wissenschaft zu leisten; Kompetenz zum Erkennen von bedeutendetechnischen Entwicklungen; Qualitätsbewusstsein, Akribie.</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 90 Std. Praktikum (Präsenzstudium)</p>		
<p>Voraussetzungen: Programmiererfahrung Modul Reinforcement Learning (INF-0207) - empfohlen</p>		
<p>Angebotshäufigkeit: jedes Sommersemester</p>	<p>Empfohlenes Fachsemester: ab dem 2.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 6</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p>		
<p>Modulteil: Praktikum Reinforcement Learning Lehrformen: Praktikum Sprache: Deutsch SWS: 6</p>		
<p>Inhalte: Die konkrete Aufgabenstellung aus dem Gebiet "Reinforcement Learning" wird jedes Semester neu entworfen.</p>		
<p>Prüfung Praktikum Reinforcement Learning praktische Prüfung Beschreibung: Ausnahmefall SoSe 2020: Praktische Prüfung</p>		

Modul INF-0272: Intelligente Signalanalyse in der Medizin <i>Intelligent Signal Analysis in Medicine</i>		5 ECTS/LP
Version 1.1.0 (seit WS17/18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Knowledge: The students learn the principal concepts of sequential signal processing, signal source separation, and feature extraction and information reduction exemplified by medically relevant audio and bio signals. They further gain insight into machine learning principles such as learning dynamics and context as is needed for many intelligent signal analysis tasks. They will learn about different problems and solutions in the analysis of a variety of signals relevant in the context of health care, wellbeing, and general medical signals analysis. Students will get to know the mindset of modern machine learning, computer-aided health care, and get to know ethical implications.</p> <p>Skills: The students will be able to choose appropriate algorithms of signal processing and machine intelligence, further develop these, design new solutions, and apply these to a broad range of medical signal analysis problems. They will practice to think logically and conceptionally in order to select appropriate solutions to a given task. Students will be able to recognise important technical developments in the field of signal processing, machine learning and e-Health/m-Health.</p> <p>Competences: The students are able to characterise, judge on the quality and suitability, and design suited algorithmic solutions for intelligent signal analysis with a focus on medical signals. They are further able to realise the learnt concepts in programs and machine learning models. Participants will be able to analyse and structure complex and practice-oriented problems in the field of m-Health and e-Health and to find suitable and state-of-the-art solutions. They know how to make scientifically meaningful evaluations of proposed systems. They will further learn how to document and present results in a reasonable and meaningful way.</p> <p>Key skills: Formal methods; Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality awareness; Scientific working; Literature research.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>30 Std. Übung (Präsenzstudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
Voraussetzungen: Knowledge of basic mathematic lectures should be present.		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Intelligente Signalanalyse in der Medizin (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Englisch		
SWS: 2		

Inhalte:

Topics: Basics of Signal Processing, Signal Source Separation, Data Acquisition and Annotation, Audio-Visual Feature Extraction, Machine Learning, e-Health, m-Health, Ethics, Python, Machine Learning Toolkits.

Literatur:

Björn Schuller: „*Intelligent Audio Analysis*“, Signals and Communication Technology, Springer, ISBN: 978-3642368059, 2013.

Modulteil: Intelligente Signalanalyse in der Medizin (Übung)

Lehrformen: Übung

Sprache: Englisch

SWS: 2

Prüfung

Intelligente Signalanalyse in der Medizin (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0273: Praktikum Mobile Sensing for Fitness and Well-being <i>Practical Module Sensing for Fitness and Wellbeing</i>		5 ECTS/LP
Version 1.0.0 (seit WS17/18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Students learn to realise concepts and models of sensing health- and fitness-related parameters on smart devices. They learn how to acquire signals from different modalities and sensors and to implement algorithms of pattern recognition on mobile devices. After participation in the Praktikum, students know how to analyse and structure complex problems in the field and to select suitable and state-of-the-art approaches to their solution.</p> <p>Participants are trained in their analytical and conceptional skills as well as in practical programming skills to transfer their knowledge to a practical task. They learn how to make scientifically meaningful assessments of their system using appropriate methods. They will know the mindset of two different fields, software development and machine learning. All knowledge obtained during the Praktikum is applied in practice-oriented tasks. Furthermore, after participation, students will be able to recognise important technical evolution in the field of sensors, mobile apps and related tools, and intelligent signal analysis.</p> <p>Students will work in teams and organise their work and task distribution in an autonomous way. They will learn how to summarise, present and document results in a reasonable way</p> <p>Key skills: Formal methods; Methods for software development and abstraction; Versioning tools; Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams, Understanding of team management; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality Awareness.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std. 60 Std. Praktikum (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Mobile Sensing for Fitness and Wellbeing		
Lehrformen: Praktikum		
Sprache: Englisch		
Angebotshäufigkeit: jedes Wintersemester		
SWS: 4		
<p>Inhalte:</p> <p>Research in the field of m-Health is focussed on the design and development of sensors, systems, and applications to recognise, interpret and simulate human states w.r.t. fitness, health, and wellbeing. In this Praktikum, students will experience in designing relevant systems, which are using modalities originating from different sensors, such as, vital signs, audio, speech, and video. In small teams, they will implement and evaluate an application running on a smart device.</p>		

Literatur:

Wird vom Dozenten / von der Dozentin bekannt gegeben

Prüfung

Praktikum Mobile Sensing for Fitness and Wellbeing

Praktikum

Modul INF-0277: Analyzing Massive Data Sets <i>Analyzing Massive Data Sets</i>		8 ECTS/LP
Version 1.2.0 (seit SoSe18) Modulverantwortliche/r: Prof. Dr. Peter Michael Fischer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an der Veranstaltung sind die Studierenden in der Lage die Konzepte und Methoden, Verfahren, Techniken und Technologien zur Analyse von massiv großen Datensätzen zu verstehen und zu bewerten. Mögliche Inhalte umfassen:</p> <ul style="list-style-type: none"> • Grundlagen des Information Retrieval • Ähnlichkeitssuche und Clustering • Analyse von Datenströmen und temporalen Daten • Webgraphen: Linkanalyse und soziale Netzwerke • Dynamische Netzwerke und Informationsausbreitung • Empfehlungssysteme und Onlinewerbung • Berechnungsverfahren für massive Datensätze <p>Außerdem können die Studierenden fachliche Lösungskonzepte zur Analyse großer Datensätze in Programme umsetzen.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken, Abwägen von Lösungsansätzen, Erwerb von Abstraktionsfähigkeiten; Fachspezifische Vertiefung; Umsetzen fachlicher Lösungskonzepte in Programme und Modelle; Kenntnisse der Vor-/Nachteile von Entwurfsalternativen und Bewertung im jeweiligen Anwendungszusammenhang; Auswahl und sichere Anwendung geeigneter Methoden; Fertigkeit, mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen anzustellen; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen;</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen: Modul Datenbanksysteme (INF-0073) - empfohlen Modul Diskrete Strukturen für Informatiker (INF-0109) - empfohlen Modul Informatik 3 (INF-0111) - empfohlen</p>		
<p>Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 6</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p> <p>Modulteil: Analyzing Massive Data Sets (Vorlesung) Lehrformen: Vorlesung Sprache: Englisch SWS: 4</p>		

Inhalte:

Die Vorlesung behandelt grundlegende Konzepte zur Analyse massiv großer Datensätze wie Informationsextraktion, Ähnlichkeitssuche, Clustering, Link- und Netzwerkanalyse sowie deren Implementierung.

Literatur:

- Mining of Massive Datasets. J. Leskovec, A. Rajaraman, J.D. Ullman. Cambridge University Press, 2014
- D. Easley, J. Kleinberg. Networks, Crowds, and Markets: Reasoning About a Highly Connected World. Cambridge University Press, 2010.
- R. Baeza-Yates, B. Ribeiro-Neto: Modern Information Retrieval

Weitere Literatur wird in der Vorlesung bekannt gegeben

Modulteil: Analyzing Massive Data Sets (Übung)

Lehrformen: Übung

Sprache: Englisch / Deutsch

SWS: 2

Prüfung

Analyzing Massive Data Sets

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0279: Music Informatics <i>Music Informatics</i>		5 ECTS/LP
Version 1.3.0 (seit SoSe18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Knowledge: The course Music Informatics presents the fundamental concepts of music theory and the music language and its representation in the visual, symbolic, and acoustic domain. Several digital formats for music symbolic representation, such as Music XML, MEI, Kern**, and MIDI protocol, as well as open source tools such as LilyPond and Csound will be introduced. Machine learning principles and techniques with applications in music information retrieval and computational musicology will be practically applied. Students will learn about different problems and solutions in the analysis of symbolic and acoustic music data. Students will get to know the mindset from both sides, the musicological and the computer scientist perspective.</p> <p>Skills: The students will understand the basic principles of music theory and its representation in digital language, being able to analyse, interpret, and create musical samples in a variety of symbolic formats and programming languages. They will learn to apply machine learning procedures, such as feature extraction and pattern recognition, to music information retrieval problems, such as key detection and music-score synchronisation, amongst other. After participation, students will know how to advance existing concepts and approaches in the field of music informatics and data analysis. Furthermore, they will be able to recognise important technical developments in the field of data science and signal processing.</p> <p>Competences: By integrating basic principles of music theory, its representation in digital language, and machine learning techniques, the students will be able to identify new problems and solutions in the field of music information retrieval considering a variety of musical styles and genres. The students are able to characterise, judge on the quality and suitability, and design suited algorithmic solutions for music data analysis in both the symbolic and the audio domain.</p> <p>Participants will be able to analyse and structure complex and practice-oriented problems in the field of music informatics and to find suitable solutions, by using state-of-the-art tools and complementary methods, if needed. They know how to make scientifically meaningful evaluations of proposed systems. They will further learn how to document and present results in a reasonable and meaningful way.</p> <p>Key skills: Computational musicology, Music theory, Digital Music Representation, Basics of Signal Processing, Machine Learning, Music Information Retrieval, Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality awareness; Scientific working; Literature research.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>30 Std. Übung (Präsenzstudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen:		
Knowledge of basic mathematic lectures should be present		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	

<p>Modulteile</p>
<p>Modulteil: Music Informatics (Lecture) Lehrformen: Vorlesung Sprache: Englisch Angebotshäufigkeit: jedes Sommersemester SWS: 2</p>
<p>Inhalte:</p> <p>In Music Informatics, the basic principles of music theory will be presented from both the traditional and computational point of view. Music will be evaluated in three domains: visual, symbolic, and acoustic; and for each of them: formats, programming languages, and machine learning tools will be studied. This course will give a basic introduction to music information retrieval and computational musicology by identify problems and solutions for different kinds of musical genres and styles.</p>
<p>Literatur:</p> <ul style="list-style-type: none"> • Meinard Müller: <i>"Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications."</i> Springer, ISBN: 978-3-319-21944-8. 2015. • Björn Schuller: <i>"Intelligent Audio Analysis"</i>, Signals and Communication Technology, Springer, ISBN: 978-3642368059, 2013.
<p>Modulteil: Music Informatics (Tutorial) Lehrformen: Übung Sprache: Englisch Angebotshäufigkeit: jedes Sommersemester SWS: 2</p>
<p>Prüfung Music Informatics (Exam) Klausur / Prüfungsdauer: 90 Minuten</p>

Modul INF-0284: Praktikum Mobile Application Development <i>Practical Module Mobile Application Development</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Students learn to realise concepts and models of mobile application development on the Android platform. They learn how to acquire data and signals from different sensors and to implement algorithms of pattern recognition and data/signal analysis on mobile devices. After participation in the Praktikum, students know how to analyse and structure complex problems in the field and to select suitable and state-of-the-art approaches to their solution.</p> <p>Participants are trained in their analytical and conceptional skills as well as in practical programming skills to transfer their knowledge to a practical task. They learn how to make scientifically meaningful assessments of their system using appropriate methods. They will know the mindset and thinking of application development and software engineering. All knowledge obtained during the Praktikum is applied in practice-oriented tasks. Furthermore, after participation, students will be able to recognise important technical evolution in the field of sensors, mobile apps and related tools, and smart devices.</p> <p>Students will work in teams and organise their work and task distribution in an autonomous way. They will learn how to summarise, present and document results in a reasonable way.</p> <p>Key skills: Formal methods; Methods for software development and abstraction; Versioning tools; Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams, Understanding of team management; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality Awareness.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std. 60 Std. Praktikum (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: Programming skills in Java are required.		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Mobile Application Development		
Lehrformen: Praktikum		
Sprache: Englisch		
SWS: 4		
<p>Inhalte:</p> <p>In the Praktikum Mobile Application Development, students will develop software for the Android platform in groups of 2-4 students.</p> <p>Students will get to know and to program several groups of Android devices, e.g., mobiles and wearables, and respective sensors. They will familiarise with the Android Studio IDE, the according toolchain and the code management tool Gi.</p> <p>Topics: Java, Android, App development, UI design, Audio Player, Map and Location, Camera control, Mobile devices, Intelligent signal processing, Wearables.</p>		

Literatur:

The literature is going to be announced by the lecturers during the course.

Prüfung

Praktikum Mobile Application Development

Praktikum

Modul INF-0291: Praktikum Selbstlernende Systeme <i>Practical Module Self-Learning Systems</i>		8 ECTS/LP
Version 1.2.0 (seit WS18/19) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Praktikum sind die Studierenden in der Lage, Problemstellungen (höherer Komplexität) in dem Bereich „maschinelle Lernverfahren“ zu verstehen und zu lösen. Sie können unterschiedliche Verfahren vergleichen und einordnen und diese eigenständig auf konkrete Beispiele aus der Praxis anwenden. Studenten können intelligente Systeme im Bezug auf die algorithmische Lösung bewerten und sind weiterhin mit Verfahren zur Leistungsevaluierung eines intelligenten Systems vertraut. Sie sind außerdem in der Lage, in kleinen Teams, größere Projektaufgaben (Entwicklung von Softwaremodulen) zu planen, nach einem selbst entwickelten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und als Team zu präsentieren. Weiterhin können sie bedeutende technische Enticklungen im Bereich „maschinelles Lernen“ erkennen und einordnen.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> - Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen, Fertigkeiten zur Entwicklung und Umsetzung von Lösungsstrategien - Kompetenz zur Vernetzung unterschiedlicher Fachgebiete - Präsentation und Dokumentation (eigener) Ergebnisse - Analytische methodische Kompetenz - Fähigkeit produktiv und zielführend im Team zu arbeiten - Akribisches Arbeiten - Fachübergreifende Kenntnisse - Systematische Weiterentwicklung von Entwurfsmodellen - Zeitmanagement - Eigenständige Literaturrecherche zu angrenzenden Themen - Grundsätze guter wissenschaftlicher Praxis 		
<p>Arbeitsaufwand: Gesamt: 240 Std. 225 Std. Praktikum (Selbststudium) 15 Std. Teilnahme an Lehrveranstaltungen (Präsenzstudium)</p>		
<p>Voraussetzungen: Empfohlen: die erfolgreiche Teilnahme an der Vorlesung OC2, Programmiererfahrung, Teamfähigkeit Modul Organic Computing II (INF-0066) - empfohlen</p>		<p>ECTS/LP-Bedingungen: Bestehen der Modulprüfung</p>
<p>Angebotshäufigkeit: jedes Semester</p>	<p>Empfohlenes Fachsemester: ab dem 2.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 2</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p> <p>Modulteil: Praktikum Selbstlernende Systeme Lehrformen: Praktikum Dozenten: Prof. Dr. Jörg Hähner Sprache: Deutsch SWS: 2 ECTS/LP: 8.0</p>		

Inhalte:

In dem Praktikum „Selbstlernende Systeme“ sollen die Studenten unterschiedliche Methoden aus dem Bereich „machine learning“ kennenlernen und vor allem selber implementieren. In einem Einführungskurs werden die benötigten theoretischen Grundlagen gelegt und diese dann innerhalb von Kleingruppen an konkreten Beispielen aus der Praxis angewandt.

Literatur:

aktuelle wissenschaftliche Paper

Zugeordnete Lehrveranstaltungen:

Praktikum zu Selbstlernende Systeme (Praktikum)

Veranstaltung wird online/digital abgehalten.

In dem Praktikum „Selbstlernende Systeme“ sollen die Studenten unterschiedliche Methoden aus dem Bereich „machine learning“ kennenlernen und vor allem selber implementieren. In einem Einführungskurs werden die benötigten theoretischen Grundlagen gelegt und diese dann innerhalb von Kleingruppen an konkreten Beispielen aus der Praxis angewandt.

Prüfung

Praktikum Selbstlernende Systeme

Portfolioprüfung, Kombination aus Praktischer und Schriftlich-Mündlicher Prüfung

Beschreibung:

schriftliche Abgaben, Softwareabnahme, Abschlussvortrag

Modul INF-0293: Advanced Deep Learning <i>Advanced Deep Learning</i>		8 ECTS/LP
Version 1.0.0 (seit WS18/19) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an dem Praxismodul verfügen die Studierenden über detailliertes und aktuelles Fachwissen auf dem Gebiet des maschinellen Lernens, können bedeutsame technische Entwicklungen identifizieren und können eine komplette Pipeline zur multimodalen Datenverarbeitung mit tiefen neuronalen Netzen implementieren. Sie können Problemstellungen und Ergebnisse des Gebiets präzise beschreiben, diskutieren und gelernte Konzepte und Methoden auf ähnliche Problemstellungen im maschinellen Lernen anwenden. Darüber hinaus analysieren die Studierenden weiterführende Konzepte, Methoden, Verfahren, Techniken und Technologien aus dem Gebiet des maschinellen Lernens, um sie in Forschungsprojekten anzuwenden, diese auf aktuelle industriennahe Aufgabenstellungen zu übertragen und dort aktiv mitzuarbeiten. Die Studierenden lernen wissenschaftlich anspruchsvolle Themenstellungen im Gebiet des maschinellen Lernens auf andere Forschungsfragen zu übertragen und darauf aufbauend ein komplexes Projekt in Gruppenarbeit auszuarbeiten. Außerdem verfügen sie über die Team- und Kommunikationsfähigkeit, um Problemstellungen auf dem Gebiet zu erörtern, Fragen und Zwischenergebnisse zu diskutieren, beschreiben und zu präsentieren. Zusätzlich sind die Studierenden in der Lage detaillierte Experimente durchzuführen und Ergebnisse zu beurteilen, vergleichen und auf Plausibilität zu prüfen.</p> <p>Schlüsselqualifikationen: Fortgeschrittene mathematisch-formale Methodik; Umsetzen fachlicher Lösungskonzepte in Programme und Modelle; Methoden zur Entwicklung größerer Softwaresysteme, Konstruktion von Abstraktionen und Architekturen; Fachübergreifende Kenntnisse; Systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten; Verstehen von Teamprozessen; Fertigkeit der Zusammenarbeit in Teams; Fähigkeit zur Leitung von Teams; Kennenlernen der Abläufe und Prozesse im Anwendungsumfeld der Informatik; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen; Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 20 Std. Vorlesung (Präsenzstudium) 40 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen: Grundkenntnisse in maschinellern Lernen und maschinellern Sehen (Grundstudiums-Vorlesungen "Multimedia Grundlagen 1" bzw. "Grundlagen der Signalverarbeitung und des Maschinellen Lernens", "Multimedia Grundlagen 2" sowie die Master-Vorlesung "Multimedia 2" bzw. "Machine Learning and Computer Vision")</p>		<p>ECTS/LP-Bedingungen: Bestehen der Portfolioprüfung</p>
<p>Angebotshäufigkeit: jedes Wintersemester</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 6</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	

Modulteile
Modulteil: Advanced Deep Learning (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2
Inhalte: <ul style="list-style-type: none">• Deep Learning in general• Deep Convolutional Neural Networks• Transfer Learning• Recurrent Neural Networks / LSTM Networks• Natural Language Processing• Multimodal Fusion (Vision+Language)• Application: Image Captioning
Zugeordnete Lehrveranstaltungen: Advanced Deep Learning (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i>
Modulteil: Advanced Deep Learning (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 4
Zugeordnete Lehrveranstaltungen: Übung zu Advanced Deep Learning (Übung) <i>*Veranstaltung wird online/digital abgehalten.*</i>
Prüfung Advanced Deep Learning Portfolioprüfung, Die Endnote setzt sich aus bewerteten Übungsblättern und einem bewerteten Teamprojekt zusammen.

Modul INF-0294: Speech Pathology <i>Speech Pathology</i>		5 ECTS/LP
Version 1.1.0 (seit WS18/19) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>Knowledge: The students learn concepts relating to signal processing, speech production, phonetics, speech and language pathology, speech analysis, feature extraction, denoising and information reduction as exemplified through the analysis of automated voice pathology detection. They further gain insight into machine learning principles, with a particular focus on deep learning solutions, as is needed to diagnose a range of different voice pathologies. They will learn about different problems and solutions in the analysis of a variety of speech, relevant in the context of health care and wellbeing.</p> <p>Skills: The students will be able to choose appropriate algorithms of signal processing and machine intelligence, further develop these, design new solutions, and apply these to the task of voice pathology detection. They will know how to analyse and structure complex problems in the field, to employ suitable approaches to solve them, and to transfer knowledge to similar tasks. After participation in the course, they will be able to implement approaches and models into programs. Students will be able to assess developed systems in a scientific way. Important technical evolution and novelties in the fields of speech analysis and medical machine learning will be recognised by them.</p> <p>Competences: The students are able to characterise, judge on the quality and suitability, and design suited algorithmic solutions for intelligent signal analysis with a focus on voice pathology detection. They are further able to present and document results in a reasonable and meaningful way.</p> <p>Key skills: Formal methods; Knowledge of advantages and disadvantages of different design alternatives; Systematical advancement of design tools; Ability to work in teams; Knowledge of workflows and processes; Ability to find solutions for practical problems; Ability to work autonomously; Quality awareness; Scientific working; Literature research.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>30 Std. Übung (Präsenzstudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen:</p> <p>Knowledge of basic mathematic lectures should be present.</p>		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Speech Pathology (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Englisch</p> <p>Angebotshäufigkeit: jedes Wintersemester</p> <p>SWS: 2</p>		

Inhalte:

The course "Pathological Speech" will give an introduction to models of speech production (e.g., source-filter models) with a focus on aspects that are relevant to pathologies and their recognition using automated methods of signal processing and machine learning. Moreover, students learn about robust feature extraction, modern methods of machine learning and machine intelligence, and implementation of such systems on devices

Topics: Speech production; Phonetics; Speech and language pathology; Signal processing; Natural language processing; Speech analysis; Feature extraction; Machine learning; Deep learning; Denoising; Information reduction; Healthcare.

Literatur:

- Björn Schuller, Anton Batliner: "Computational Paralinguistics: Emotion, Affect and Personality in Speech and Language Processing", Wiley, ISBN: 978-1119971368, 2013.
- Further literature is going to be announced during the lecture.

Zugeordnete Lehrveranstaltungen:

Speech Pathology (Vorlesung)

Veranstaltung wird online/digital abgehalten.

In this novel course, students will learn about human speech production and related pathologies. The course will introduce recent source-filter models and the principal concepts of signal processing, feature extraction, and information reduction. State-of-the-art machine learning techniques will be discussed as well as implications of practical applications of speech-related health care technologies.

Modulteil: Speech Pathology (Übung)

Lehrformen: Übung

Sprache: Englisch

Angebotshäufigkeit: jedes Wintersemester

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Speech Pathology (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Speech Pathology

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0296: Praktikum Interactive Machine Learning <i>Practical Module Interactive Machine Learning</i>		8 ECTS/LP
Version 1.0.0 (seit WS18/19) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierenden mit Grundkonzepten des interaktiven maschinellen Lernens vertraut. Sie sind in der Lage, fachliche Lösungskonzepte in Modelle zu übersetzen und beherrschen die Auswahl und sichere Anwendung geeigneter Methoden. Darüber hinaus erhalten sie einen Einblick in aktuelle Arbeiten des Forschungsgebiets. Weiterhin werden Kompetenzen in den Bereichen Team- und Kommunikationsfähigkeit sowie Selbstorganisation durch die Bearbeitung von Projektaufgaben und deren eigenständige Projektplanung vermittelt. Im Rahmen dieser Projektaufgaben werden Kenntnisse der Denkweise und Sprache des Forschungsgebietes durch die Diskussion und Präsentation der Projektergebnisse gefördert.</p> <p>Schlüsselqualifikationen: Umsetzen fachlicher Lösungskonzepte in Programme und Modelle; Fachspezifische Vertiefung; Fachübergreifende Kenntnisse; Fertigkeit, mit geeigneten Methoden wissenschaftlich aussagekräftige Bewertungen anzustellen; Kenntnisse der Denkweise und Sprache anwendungsrelevanter Disziplinen; Fertigkeit der Zusammenarbeit in Teams; Kenntnisse von praxisrelevanten Aufgabenstellungen; Kennenlernen der Abläufe und Prozesse im Anwendungsumfeld der Informatik; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Kompetenz zum Erkennen von bedeutenden technischen Entwicklungen</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Praktikum (Präsenzstudium) 150 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: Programmiererfahrung		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile
<p>Modulteil: Praktikum Interactive Machine Learning Lehrformen: Praktikum Sprache: Deutsch SWS: 6</p>
<p>Zugeordnete Lehrveranstaltungen: Praktikum Interactive Machine Learning (Praktikum) <i>*Veranstaltung wird online/digital abgehalten.*</i> Dieser Kurs fokussiert sich auf "interaktives Maschinelles Lernen" (iML). Wir definieren iML als Maschinelles Lernen bei dem der Mensch als zentraler Teil, interaktiv an verschiedenen Stellen in den Lernprozesses eingreift. Er überwacht dabei die Ergebnisse der Maschine und liefert Eingaben und Korrekturen um den Lernprozess zu verbessern. Während traditionelle ML Systeme menschliche Eingaben auf das Bereitstellen von Annotationen beschränken, beschäftigt sich iML damit Interaktionsdesign Richtlinien für ML Systeme zu erschaffen, und neue Methoden zu entwickeln menschliche Expertise in den ML Prozess einfließen zu lassen. Dadurch werden Systeme transparenter und nachvollziehbarer, was sich mit dem aktuell aufstrebenden Forschungstrend der "explainable AI" (XAI) überschneidet. Der Kurs besteht aus mehreren praktischen Hands-on Projekten und einer Einführung in verschiedene Themen rund um das Thema "interactive Machine Learning". Grundkenntnisse in Python sind erwünscht.</p>

... (weiter siehe Digicampus)

Prüfung

Praktikum Interactive Machine Learning

praktische Prüfung

Modul INF-0297: Praktikum Prozessorbau <i>Practical Module Processor Design</i>		5 ECTS/LP
Version 2.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer Dr. Martin Frieb		
<p>Lernziele/Kompetenzen: Die Studierenden erwerben Kompetenzen in den folgenden Bereichen auf einem fortgeschrittenen, praxisorientierten, aber wissenschaftlichen Niveau: Entwurfsprozess für digitale Schaltungen, Schaltungslogik und Gatter, physikalische Grundlagen der elektronischen Bauteile, Beschreibung von Hardware mit einer Hardwarebeschreibungssprache. Zunächst lernen die Studierenden die Verknüpfung logischer Gatter und bauen einen Halb- und einen Volladdierer. Sie verstehen den Entwicklungsprozess für digitale Schaltungen und wenden diesen direkt praktisch an, indem sie ihren eigenen RISC-V-Prozessor entwickeln. Sie modellieren und implementieren ihn selbstständig mit Hilfe der Hardwarebeschreibungssprache VHDL. Dazu lernen sie die Vor- und Nachteile von schematischer und textueller Hardware-Beschreibung kennen und können entscheiden, wann es sinnvoll ist, welche Variante einzusetzen. Darüber hinaus kombinieren sie synchrone und asynchrone Prozesse, um ein gutes Zusammenspiel der Komponenten ihres selbstgebauten Mikroprozessors zu erreichen. Schließlich beurteilen die Studierenden die Effizienz ihrer Implementierung anhand der erzielten Taktfrequenz und des benötigten Hardware-Aufwandes. In einer abschließenden Projektphase lernen sie, eine komplexe Aufgabenstellung zu planen, nach einem selbst entwickelten fundierten Projektplan zu lösen und die Resultate angemessen im Plenum zu diskutieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zur verständlichen Darstellung und Dokumentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein, Akribie; Projektgebundene Arbeit und Zeitmanagement; Auswahl und sichere Anwendung geeigneter Methoden; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Selbstreflexion</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Praktikum (Präsenzstudium)</p>		
<p>Voraussetzungen: Modul Systemnahe Informatik (INF-0138) - empfohlen Modul Prozessorarchitektur (INF-0147) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Modulteil: Praktikum Prozessorbau Lehrformen: Praktikum Sprache: Deutsch SWS: 4</p>		
<p>Inhalte: Die Veranstaltung "Prozessorbau" stellt Methoden des logischen Entwurfs digitaler Schaltungen dar, angefangen von der abstrakten Beschreibung in einer Hardwarebeschreibungssprache (wie VHDL) bis hin zur physikalischen Implementierung auf Transistorebene. Im praktischen Teil der Veranstaltung wird die Hardwareentwicklung am Beispiel einer fünfstufigen Prozessorpipeline veranschaulicht. Das Ergebnis ist ein in VHDL entwickelter lauffähiger Prozessor für ein FPGA-Prototypenboard</p>		

Literatur:

- Uwe Brinkschulte, Theo Ungerer, Mikrocontroller und Mikroprozessoren, Springer Verlag, Heidelberg, dritte Auflage 2010
- John L. Hennessy, David A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 5. Auflage, 2011

Prüfung

Praktikum Prozessorbau

praktische Prüfung

Modul INF-0307: Modellbasierte Entwicklung und Analyse von Software Systemen <i>Model-Based Development and Analysis of Software Systems</i>		6 ECTS/LP
Version 1.1.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Modellbasierte Entwicklung und Analyse von Software Systemen befasst sich mit der Effizienzsteigerung in der Softwareherstellung durch Automatisierung und Wiederverwendung. Die Teilnehmer lernen in der Veranstaltung Methoden zur modellgetriebenen Entwicklung von Softwaresysteme anzuwenden und zu vergleichen. Sie entwickeln dabei vertiefte, fachspezifische Lösungskonzepte für MDSD. Sie können aktuelle Technologien und Standards für MDSD bewerten und analysieren deren Anwendbarkeit in praxisrelevanten Aufgabenstellungen. Die Teilnehmer bauen Fertigkeiten zur Analyse und Strukturierung komplexer Informatikproblemstellungen bei der Generierung von Infrastrukturcode, Subsysteme, Konfigurationen oder ganze Anwendungen aus Modellen auf. Dabei entwickeln sie Fertigkeiten zum logischen, analytischen und konzeptionellen Denken und können Problemlösungen systematisch entwickeln und beurteilen.</p> <p>Schlüsselqualifikation: Fachübergreifende Kenntnisse; Kompetenz zur Vernetzung unterschiedlicher Fachgebiete; Team- und Kommunikationsfähigkeit; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein; Kennenlernen der Abläufe und Prozesse im Anwendungsumfeld der Informatik; Formale quantitative Grundlagen kennen und verstehen; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen</p>		
<p>Arbeitsaufwand: Gesamt: 180 Std. 23 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 22 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 45 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: Frühere Veranstaltung "Modellgetriebene Softwareentwicklung" darf nicht belegt worden sein wegen Überschneidungen.		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 5	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Modellbasierte Entwicklung und Analyse von Software Systemen (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 3		
Inhalte: Modellbasierte Entwicklung und Analyse von Software Systemen befasst sich mit der Effizienzsteigerung in der Softwareherstellung durch Automatisierung und Wiederverwendung. Dabei werden Infrastrukturcode, Subsysteme, Konfigurationen oder ganze Anwendungen aus Modellen generiert.		

Literatur:

- Folien
- Pohl et al. Software Product Line Engineering: Foundations, Principles, and Techniques
- Kleppe et al: MDA explained
- Hitz et al: UML@Work
- weitere Literatur in der Vorlesung zu speziellen Themen

Modulteil: Modellbasierte Entwicklung und Analyse von Software Systemen (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Modellbasierte Entwicklung und Analyse von Software Systemen

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0308: Software-intensive Systeme <i>Software-intensive Systems</i>		6 ECTS/LP
Version 1.1.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Die Studierenden sind in der Lage SW-Architekturen zu erstellen (K3), zu bewerten (K6) und zu dokumentieren. Dafür können sie fachliche Lösungskonzepte in Modelle überführen und kennen Methoden zur Entwicklung solcher Abstraktionen und Architekturen. Sie können Vor- und Nachteile von Entwurfsalternativen beschreiben (K4) und können diese im jeweiligen Anwendungszusammenhang bewerten (K6). Probleme können dabei eigenständig erkannt (K4) und Lösungen systematisch entworfen (K5) und realisiert (K3) werden. Weiterhin haben sie Fertigkeiten für die Analyse und Strukturierung der Problemstellungen in Unternehmensarchitekturen entwickelt und kennen die Konzepte und Vorgehensweisen für die Erstellung solcher Architekturen. Die Studierenden können praxisrelevanten Fragestellungen in Unternehmensarchitekturen benennen (K1). Sie können geeignete Methoden zur Architekturerstellung und -bewertung auswählen und sicher anwenden. Die Studenten kennen Modellierungssprachen und Patterns zur Erstellung von Software- und Unternehmensarchitekturen. Sie haben die Kompetenz bedeutende technische Entwicklungen zu erkennen.</p> <p>Schlüsselqualifikation: Kompetenz zur Vernetzung unterschiedlicher Fachgebiete; Team- und Kommunikationsfähigkeit; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Praxiserfahrung und Berufsbefähigung</p>		
<p>Arbeitsaufwand: Gesamt: 180 Std. 22 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 23 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 45 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Frühere Veranstaltung "Softwarearchitekturen und Enterprise Architecture Management" darf nicht belegt worden sein wegen Überschneidungen.		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 5	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Software-intensive Systeme (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 3		
Inhalte: Der Vorlesungsinhalt umfasst Patterns, Modellierungstechniken und die Evaluation von Softwarearchitekturen. Weiterhin wird auf den Bereich des Enterprise Architecture Managements eingegangen.		
Literatur: <ul style="list-style-type: none"> • Bass et al: Software Architecture in Practice • Clements et al: Documenting Software Architectures • Clements et al: Evaluation of Software Architectures • Kopetz: Real-Time Systems 		
Zugeordnete Lehrveranstaltungen:		

Software-intensive Systeme (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Inhalte: - Was sind SW-Architekturen? - Pattern und Muster für SW-Architekturen - Modellierung von SW-Architekturen - Evaluation von SW-Architekturen - Eingebettete Systeme: Definitionen, Anforderungsanalyse, Modellierung, Architektur - Software-Qualität: Definitionen und Standards, Funktionstest, Überdeckungsmaße, HiL-, Integrations- und Abnahmetests, Verifikation und Validierung, Architecture Design and Reliability - Enterprise Architecture Management: Methoden, Frameworks, Tools

Modulteil: Software-intensive Systeme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Software-intensive Systeme (Übung)

Veranstaltung wird online/digital abgehalten.

Der Übungsbetrieb soll die Inhalte der Vorlesung veranschaulichen und wird aus der gruppenweisen Vorstellung praktischer Beispiele von Architekturen und Frameworks bestehen. Die konkreten Themen folgen.

Prüfung

Software-intensive Systeme

Mündliche Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0309: Echtzeitsysteme <i>Real-Time Systems</i>		8 ECTS/LP
Version 1.5.0 (seit WS19/20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen:</p> <p>Die Vorlesung vermittelt grundlegende und weiterführende Kenntnisse zu Echtzeitsystemen wie sie in nahezu allen eingebetteten Systemen, jedoch speziell in den Bereichen Automobil, Luft-/Raumfahrt und Robotik vorkommen. Die theoretischen Grundlagen werden sich am aktuellen Forschungsstand orientieren und eine weitere Beschäftigung der Studierenden mit dem Thema eingebettete Echtzeitsysteme auf wissenschaftlichen Niveau ermöglichen.</p> <p>Die Vorlesung wird den Studierenden die Fähigkeit vermitteln verschiedene eingebettete Systeme anhand ihrer Echtzeitanforderungen zu unterscheiden und einzuordnen. Die Studierenden lernen aktuelle Methoden zur Validierung des Zeitverhaltens anzuwenden, zu vergleichen, und im Hinblick auf eine mögliche Zertifizierung des Zeitverhaltens kritisch zu analysieren. Dies beinhaltet die Optimierung und Auswahl von Echtzeit-Schedules und deren Verifikation. Die Vorlesung behandelt zudem verschiedene Prozessortypen, und wird näher auf die Besonderheiten von Einkern und Mehrkern-Prozessoren im Echtzeitbereich eingehen. Die Studierenden werden hierbei in die Lage versetzt, Prozessoren anhand ihrer Eignung für Echtzeitsysteme zu klassifizieren und den Einfluss von Entwurfsentscheidungen auf das Echtzeitverhalten und die Analyse des Echtzeitverhaltens zu untersuchen. Der Lehrstoff wird anhand von Fallstudien aus den Bereichen Automobil und Luftfahrt exemplifiziert und von den Studierenden anhand eines einfachen Echtzeitsystems angewendet.</p> <p>Schlüsselqualifikationen:</p> <p>Analytisch-methodische Kompetenz, Abwägung von Lösungsansätzen, Präsentation von Lösungen von Übungsaufgaben; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein, Akribie; Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 240 Std.</p> <p>30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>60 Std. Vorlesung (Präsenzstudium)</p> <p>30 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen:</p> <p>keine</p>		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Echtzeitsysteme (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Deutsch / Englisch</p> <p>SWS: 4</p>		

Inhalte:

- WCET Analyse
- Scheduling Analyse
- Programmierung von Echtzeitsystemen
- Prozessoren für Echtzeitsysteme
- Echtzeitbetriebssysteme
- Zertifizierung von Echtzeitsystemen

Literatur:

- Sanjoy Baruah, Marko Bertogna, Giorgio Buttazzo, Multiprocessor Scheduling for Real-Time Systems, Springer, 2015.
- Giorgio Buttazzo, Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications, Springer, 2011.
- Heinz Wörn, Uwe Brinkschulte, Echtzeitsysteme, Springer Verlag, Berlin/Heidelberg, 2005
- Uwe Brinkschulte, Theo Ungerer, Mikrocontroller und Mikroprozessoren, Springer Verlag, Heidelberg, dritte Auflage 2010

Zugeordnete Lehrveranstaltungen:

Echtzeitsysteme (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Echtzeitsysteme sind Computer Systeme, deren Korrektheit nicht nur über das korrekte Ergebnis einer Berechnung, sondern auch über das korrekte Zeitverhalten definiert wird. Vereinfacht gesagt: Wenn ein eingebettetes System zu spät reagiert, ist es genauso unbrauchbar, wie wenn es gar nicht oder falsch reagiert hätte. Ein Airbag zum Beispiel sollte aufgehen, bevor der Kopf des Fahrers auf das Lenkrad knallt. Ein selbstfahrendes Fahrzeug muss ebenso rechtzeitig autonom bremsen, bevor ein Unfall geschehen ist, und eine Kaffeemaschine sollte aufhören Kaffee auszuschenken, bevor die Tasse überläuft. In dieser Vorlesung beschäftigen wir uns mit dem Zeitverhalten eben solcher eingebetteter Echtzeitsysteme. Wir lernen wie wir das Zeitverhalten dieser Systeme beeinflussen und validieren können und wir lernen, welchen Einfluss verschiedene Entwurfsentscheidungen auf das Zeitverhalten haben.

... (weiter siehe Digicampus)

Modulteil: Echtzeitsysteme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Echtzeitsysteme (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Echtzeitsysteme

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0310: Perlen der Algorithmik <i>Algorithmic Gems</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an der Vorlesung haben die Studierenden einen guten Überblick über mehrere Teilgebiete der Algorithmik. Sie kennen verschiedene Berechnungsmodelle und verschiedene Arten der Bewertung und der Analyse von Datenstrukturen und Algorithmen und können diese auf neue Problemstellungen anwenden. Sie sind vertraut mit Algorithmen und Datenstrukturen, die nicht zum Standardkanon eines Informatikstudiums gehören.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fähigkeit zur präzisen fachlichen Ausdrucksweise; Beherrschung effektiver Lern- und Arbeitstechniken; Abstraktionsfähigkeit; Problemlösungskompetenz; Qualitätsbewusstsein.</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vorlesung (Präsenzstudium)</p>		
<p>Voraussetzungen: Teilnehmer der Vorlesung sollten über gute algorithmische Kenntnisse verfügen. Modul Informatik 3 (INF-0111) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Perlen der Algorithmik (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Die Vorlesung stellt eine Reihe meist unabhängiger Algorithmen und Datenstrukturen vor und versucht so, den Teilnehmern einen Überblick über Sichtweisen, Ziele, Methoden und Ergebnisse vieler Teilgebiete der Algorithmik zu vermitteln. Die konkrete Themenauswahl basiert auf Kriterien wie theoretische oder praktische Wichtigkeit, relative Einfachheit, Eleganz und Neuheit.</p>		
<p>Literatur: Ein englischsprachiges Skript wird zur Verfügung gestellt.</p>		
Modulteil: Perlen der Algorithmik (Übung)		
Lehrformen: Übung		
Sprache: Deutsch		
SWS: 2		

Prüfung

Perlen der Algorithmik

Mündliche Prüfung, Dauer: 30-45 Minuten

Modul INF-0315: Deep Learning <i>Deep Learning</i>		5 ECTS/LP
Version 1.2.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>The course Deep Learning covers the historical and formal fundamentals of Neural Networks, as well as the core principles of Machine Learning and data modelling.</p> <p>Upon completing the course, students will have the skills and knowledge to be able to choose suitable approaches and network architectures for specific tasks and know the pros and cons of design alternatives, as assessed in the respective application context. They will be able to apply and implement the discussed technical concepts in programs and systems. Furthermore, they will have the ability to analyse Deep Neural Network-based models and to design novel architectures and training methods.</p> <p>During the course, the participants will improve their skills in logical, analytical, and conceptual thinking. Students will gain the ability to make scientifically meaningful assessments in the field of machine learning and data science using appropriate methods. They will get used to the way of thinking and the language of relevant disciplines.</p> <p>Moreover, students will gain the ability to, convincingly, present their developed ideas and concepts. They will be able to apply their new knowledge to practical tasks and solve many real-life problems through the appropriate application of machine learning. They will also develop the competence to identify significant technical developments in the field.</p> <p>Key qualifications: analytical skills, data science cross-disciplinary knowledge, procedures and processes in creating practical systems, ability to present and document results in a comprehensible way, skill to solve problems under practical conditions, self-reflection, quality awareness, meticulousness, teamwork</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>30 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen:</p> <p>Knowledge of basic mathematic lectures should be present.</p>		
<p>Angebotshäufigkeit: in der Regel mind. 1x pro Studienjahr</p>	<p>Empfohlenes Fachsemester: ab dem 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 4</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Moduleile</p>		
<p>Modulteil: Deep Learning (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Englisch</p> <p>SWS: 2</p>		
<p>Inhalte:</p> <p>Perceptron, Feed-forward Neural Networks, Gradient-based Learning, Backpropagation, Recurrent Neural Networks, Convolutional Neural Networks, Autoencoders, Transfer Learning, Generative Adversarial Nets, Attention, Connectionist Temporal Classification, Data Preprocessing, Evaluation, Audio Classification, Object Detection, Natural Language Processing</p>		
<p>Literatur:</p> <p>Ian Goodfellow; Yoshua Bengio; Aaron Courville (2016). <i>Deep Learning</i>. Cambridge, Massachusetts: MIT Press.</p> <p>Further literature is going to be announced during the lecture.</p>		

Modulteil: Übung zu Deep Learning

Lehrformen: Übung

Sprache: Englisch

SWS: 2

Prüfung

Deep Learning

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0316: Machine Learning and Computer Vision <i>Machine Learning and Computer Vision</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul besitzen die Studierenden fortgeschrittene Kenntnisse des maschinellen Lernens (Entscheidungsbäume, neuronale Netze und tiefe neuronale Netze, Hypothesenevaluation, Instanz-basiertes Lernen, Bayessches Lernen, Lerntheorie), der Datenreduktion (z.B. Hauptkomponentenanalyse), der fortgeschrittenen Bildverarbeitung und des maschinellen Sehens und können diese anwenden. Sie sind in der Lage, wissenschaftlich komplexe Verfahren auf dem Gebiet der multimedialen Datenverarbeitung zu analysieren, zu verstehen und programmatisch umzusetzen, sowie die erlernten Prinzipien auf neue Fragestellungen geeignet anzuwenden. Sie entwickeln Fertigkeiten zum logischen, analytischen und konzeptionellen Denken im Bereich der digitalen Signalverarbeitung und multimedialen Datenverarbeitung.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Logik; umsetzen fachlicher Lösungskonzepte in Modelle; fachübergreifende Kenntnisse; Entwicklung und Umsetzung von Lösungsstrategien komplexer Problemstellungen; systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Machine Learning and Computer Vision (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 4		
<p>Inhalte: Die Vorlesung gibt einen guten Überblick über alle Aspekte des maschinellen Lernens und der maschinellen Extraktion von Informationen aus Multimediadaten (z.B. "Google Image Search", "Google Goggles"). Die erlernten Konzepte werden in den Übungen anhand von erfolgreichen Beispielen aus der Praxis ausprobiert, geübt, analysiert und bewertet. Zum Ende des Semesters werden fortgeschrittene Themen wie Objektdetektion und Objekterkennung von Gesichtern und Menschen behandelt. Die Inhalte der Vorlesung umfassen: Machine Learning (Decision Tree Learning, Artificial Neural Networks, Bayesian Learning, Discrete Adaboost), Data Reduction (Quantization (K-Means Clustering, Affinity Propagation), Dimensionality Reduction Techniques (PCA, NMF, Random Projection, MDS)) und Image Processing & Computer Vision (Salient Feature Points and Feature Descriptors, Object Detection (Face/Car/People Detection), Object Recognition (Face Recognition), Image Search with pLSA)</p>		

Literatur:

Literaturhinweise werden zu Beginn des Semesters bekanntgegeben.

Modulteil: Machine Learning and Computer Vision (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Machine Learning and Computer Vision (Klausur)

Klausur / Prüfungsdauer: 120 Minuten

Beschreibung:

Die Prüfung kann jedes Semester in der Prüfungszeit abgelegt werden.

Modul INF-0319: Praktikum Interdisziplinäres Projekt Ingenieurinformatik <i>Interdisciplinary Project Engineering Informatics</i>		6 ECTS/LP
Version 1.0.0 (seit WS19/20) Modulverantwortliche/r: Prof. Dr.-Ing. Johannes Schilp Prof. Dr.-Ing. Lars Mikelsons, Prof. Dr.-Ing. Christoph Ament		
Lernziele/Kompetenzen: Die Studierenden beschäftigen sich mit einer fächerübergreifenden Aufgabenstellung aus dem Bereich der Produktionsinformatik, Regelungstechnik und Mechatronik. Sie setzen komplexe Verfahren und Techniken ein, die teilweise in den einzelnen Vorlesungen bereits theoretisch behandelt wurden. Das praxisnahe Projekt orientiert sich an einer Studenten-Challenge, wie beispielweise der Sioux Mechatronics Trophy oder dem James Dyson Award, und wird in Kleingruppen bearbeitet. Der Anwendungsfall erfordert die Bewertung und Übertragung der Konzepte und Methoden sowie deren interdisziplinäre Kombination. Der zeitliche Ablauf dieses Praktikums wird in Digicampus bekannt gegeben, ebenso wie die spezifische Aufgabenstellung der Challenge einschließlich der Voraussetzungen und der Lehrstuhleteiligungen. Schlüsselqualifikationen: Team- und Kommunikationsfähigkeit, strukturiertes und gewissenhaftes Arbeiten, anwendungsorientierte Problemlösung, Ergebnisbewertung und -dokumentation, Abwägen von Lösungsansätzen, Fertigkeit zum logischen, analytischen und konzeptionellen Denken, Abstraktionsfähigkeit		
Arbeitsaufwand: Gesamt: 180 Std. 120 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Praktikum (Präsenzstudium)		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Praktikum Interdisziplinäres Projekt Ingenieurinformatik Lehrformen: Praktikum Sprache: Deutsch SWS: 4 ECTS/LP: 6.0		
Inhalte: Die Studenten bearbeiten in Kleingruppen anwendungsorientierte Aufgaben zu Themenbereichen im industriellen Umfeld.		
Literatur: wird im jeweiligen Semester bekannt gegeben.		
Prüfung Praktikum Interdisziplinäres Projekt Ingenieurinformatik praktische Prüfung		

Modul INF-0335: Safety-Critical Systems <i>Safety-Critical Systems</i>		5 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen:</p> <p>Die Studierenden lernen die Fehlertoleranzanforderungen in eingebetteten Systemen kennen und ordnen diese anhand der zugrundeliegenden Vorschriften und Normen (z.B. ISO 26262) zum Entwurf sicherheitsrelevanter Systeme ein.</p> <p>Sie können die Ursachen und Effekte von Hardware-Fehlern beschreiben und sind in der Lage, unterschiedliche Redundanztechniken zu analysieren und zu bewerten.</p> <p>Weitere Schwerpunkte der Vorlesung sind die Anwendung der stochastischen Grundlagen der Fehlerrechnung, das Analysieren und Modellieren von Rechensystemen mit Zuverlässigkeitsblockdiagrammen und Fehlerbäumen, sowie das Differenzieren unterschiedlicher Redundanzarten und deren Verwendung in verschiedenen Hard- und Software-Techniken zur Fehlererkennung.</p> <p>Außerdem werden Techniken zur Fehlerkorrektur, Fehlerdiagnose und Rekonfiguration vorgestellt und untersucht. Die Studierenden lernen verschiedene Konzepte fehlertolerierender Rechensysteme aus Theorie und Praxis zu klassifizieren und zu vergleichen, wie z.B. Lockstep-Ausführung, Prozess-Redundanz, ECC-Speicher-Implementierungen oder Kontrollflussüberprüfung.</p> <p>In den praktischen Übungen wenden die Studierenden die Methoden der Modellierung und Fehlerrechnung an verschiedenen Aufgaben an, arbeiten ihre Lösungen aus und vergleichen auf qualitativer Ebene verschiedene Fehlertoleranzkonzepte aus Forschung und Praxis.</p> <p>Schlüsselqualifikationen: Analytisch-methodische Kompetenz in der Analyse fehlertolerierender Rechensysteme, Abwägung von Lösungsansätzen, Verständliche Präsentation von Lösungen von Übungsaufgaben, Fähigkeit, vorhandenes Wissen selbstständig zu erweitern, Selbstreflexion</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Safety-Critical Systems (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Deutsch</p> <p>SWS: 2</p>		

Inhalte:

Die Vorlesung betrachtet Methoden zur Einhaltung der funktionalen Sicherheit in Rechensystemen, sowie deren Entwurf und Analyse. Zunächst werden verschiedene Fehlerarten charakterisiert und die Bedeutung von Fehlermodellen hervorgehoben. Danach werden unterschiedliche Hardware- und Software-Methoden zur Erkennung und Tolerierung von Fehlern vorgestellt. Die diskutierten Maßnahmen beziehen sich nicht nur auf strukturelle, sondern auch auf zeitliche und informationelle Redundanz (fehlertolerierende Codes). Nach einem kurzen Repetitorium der Wahrscheinlichkeitsrechnung und Kombinatorik werden verschiedene Analysemethoden wie klassische Wahrscheinlichkeitsrechnung, Zuverlässigkeits-Blockdiagramme, Markovketten, FMEA und Fehlerbäume vorgestellt, Unterschiede hervorgehoben und anhand praktischer Beispiele erläutert. Verschiedene Methoden zur Fehlerinjektion, die zur Bewertung von fehlertolerierenden Systemen notwendig sind, werden kurz besprochen.

Literatur:

- D. Sorin: Fault Tolerant Computer Architecture, Morgan and Claypool, 2009
- S. Mukherjee: Architecture Design for Soft Errors, Morgan Kaufmann, 2008
- I. Koren, C.M. Krishna: Fault-Tolerant Systems, Morgan Kaufmann, 2007

Modulteil: Safety-Critical Systems (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Safety-Critical Systems

Klausur / Prüfungsdauer: 60 Minuten

Modul INF-0353: Praktikum Programmieren in der biomedizinischen Informatik (Master) <i>Practical Module Biomedical Programming (Master)</i>		6 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Frank Kramer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Praktikum Programmieren in der Biomed. Informatik verstehen die Studierenden praxisnahe Problemstellungen höherer Komplexität im Bereich Softwareentwicklung und Auswertungen in den Anwendungsbereichen der biomedizinischen Informatik. Die Studierenden erlangen tiefere fachspezifische als auch fächerübergreifende Kenntnisse und Fähigkeiten, beispielsweise Bioinformatik, Medizininformatik und Statistik. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren, zu präsentieren und verständlich zu dokumentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewusstsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Projektmanagementfähigkeiten</p>		
<p>Bemerkung: Wenn Sie bereits das Modul "INF-0326: Praktikum Programmieren in der biomedizinischen Informatik (Vertiefung)" gehört haben, ist eine erneute Einbringung nicht möglich!</p>		
<p>Arbeitsaufwand: Gesamt: 180 Std. 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 90 Std. Praktikum (Präsenzstudium) 30 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
<p>Moduleil: Praktikum Programmieren in der biomedizinischen Informatik (Master) Lehrformen: Praktikum Sprache: Deutsch SWS: 6</p>		

Inhalte:

Die Teilnehmer des Bio-/Medizininformatik Programmierpraktikums erlernen, wie man kleinere Anwendungs- und Infrastrukturprogramme im biologischen und medizinischen Kontext implementiert.

Ziel des Praktikums ist es, dass die Studierenden diverse kleinere Programme in Python zu medizinischen Fragestellungen entwickeln.

Die Studierenden erwerben anhand von täglichen Arbeitsaufgaben Grundkenntnisse über:

- Python Programmierung
- Nutzung von Bio-/Medizinischen Datenbanken
- Datentransformation im biologischen und medizinischen Kontext
- Krankheitsforschung in der Bio-/Medizininformatik
- Hochdurchsatzdaten Analyse

Das Praktikum wird als 2-wöchige Blockveranstaltung in den Semesterferien angeboten, besteht aus einer kurzen täglichen Einführung zu den aktuellen Arbeitsaufgaben und der anschließenden selbstständigen Implementierung durch die Studierenden.

Während des selbstständigen Arbeitens wird zu festgelegten Zeiten einer Betreuer für Hilfestellungen und Fragen anwesend sein.

Während des Semesters wird es ein einmaliges Treffen aus organisatorischen Gründen geben.

Prüfung

Praktikum Programmieren in der biomedizinischen Informatik (Master)

praktische Prüfung / Prüfungsdauer: 30 Minuten

Modul INF-0367: Advanced Machine Learning and Computer Vision		5 ECTS/LP
Version 1.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach der erfolgreichen Teilnahme an diesem Modul besitzen die Studierenden vertiefte fortgeschrittene Kenntnisse des maschinellen Lernens (Support Vector Machines und tiefe neuronale Netze und deren Grundbausteine) und des maschinellen Sehens (tiefe neuronale Netzarchitekturen und Systeme) und können diese anwenden. Sie sind in der Lage, wissenschaftlich komplexe Verfahren auf dem Gebiet der Bild-, Text-, Video- und Signalverarbeitung zu analysieren, zu verstehen und programmatisch umzusetzen, sowie die erlernten Prinzipien auf neue Fragestellungen geeignet anzuwenden. Sie entwickeln Fertigkeiten zum logischen, analytischen und konzeptionellen Denken im Bereich des maschinellen Lernens und Sehens.</p> <p>Schlüsselqualifikationen: fortgeschrittene mathematisch-formale Logik; kritisches Lesen und Analysieren von wissenschaftlichen Veröffentlichungen; Umsetzen fachlicher Lösungskonzepte in Modelle; fachübergreifende Kenntnisse; Entwicklung und Umsetzung von Lösungsstrategien komplexer Problemstellungen; systematische Weiterentwicklung von Entwurfsmethoden; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen</p>		
<p>Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p>		
<p>Voraussetzungen: Kenntnisse in maschinellern Lernen und maschinellern Sehen (Master-Vorlesung INF-0092 "Multimedia II" bzw. INF-0316 "Machine Learning and Computer Vision")</p>		<p>ECTS/LP-Bedingungen: Bestehen der Modulprüfung</p>
<p>Angebotshäufigkeit: jedes Wintersemester</p>	<p>Empfohlenes Fachsemester: ab dem 2.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 4</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p>		
<p>Modulteil: Advanced Machine Learning and Computer Vision (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 2</p>		
<p>Inhalte: Die Vorlesung gibt einen vertieften Einblick in alle Aspekte des maschinellen Lernens und des maschinellen Sehens. Die erlernten Konzepte werden in den Übungen anhand von erfolgreichen Beispielen aus der Praxis ausprobiert, geübt, analysiert und bewertet. Die Inhalte der Vorlesung umfassen: Support Vector Machines, Grundbausteine von tiefen Neuronalen Netzen (Layerstrukturen, Normalisierung, Attention-Mechanismen), sowie aktuelle Referenzarchitekturen und -system für Bild-, Text-, Videoverarbeitung und deren Kombination mit weiteren Sensorsignalen.</p>		
<p>Literatur: Literaturhinweise werden zu Beginn des Semesters bekanntgegeben.</p>		
<p>Zugeordnete Lehrveranstaltungen:</p>		

Advanced Machine Learning and Computer Vision (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Modulteil: Advanced Machine Learning and Computer Vision (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Advanced Machine Learning and Computer Vision (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Advanced Machine Learning and Computer Vision (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Beschreibung:

Die Prüfung kann jedes Semester in der Prüfungszeit abgelegt werden.

Modul INF-0368: Embedded Hardware <i>Embedded Hardware</i>		5 ECTS/LP
Version 1.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen:</p> <p>Die Vorlesung vermittelt grundlegende und weiterführende Kenntnisse zu Hardwarekomponenten (z.B. GPIO, Hardware-Timer, ...) wie sie in nahezu allen eingebetteten Systemen vorkommen. Außerdem werden die Funktionsprinzipien von verschiedenen Sensoren und Aktuatoren vorgestellt. Ein besonderer Fokus liegt dabei auf deren Einsatz zur Interaktion mit der physikalischen Welt und auf Besonderheiten bei deren Anwendung. Die theoretischen Grundlagen werden sich am aktuellen Forschungsstand orientieren und eine weitere Beschäftigung der Studierenden mit dem Thema eingebetteter Hardware auf wissenschaftlichen Niveau ermöglichen.</p> <p>Die Vorlesung wird den Studierenden die Fähigkeit vermitteln verschiedene eingebettete Systeme anhand ihrer Hardwareanforderungen zu unterscheiden und einzuordnen. Die Studierenden lernen aktuelle Methoden zur Programmierung der Komponenten anzuwenden, zu vergleichen, und im Hinblick auf eine effiziente Nutzung kritisch zu analysieren. Dies beinhaltet die Auswahl und Beurteilung von für das System relevanten Hardwarekomponenten, sowie die Entwicklung einer für ein bestehendes Problem optimierten Konfiguration der Komponenten. Die Vorlesung behandelt zudem verschiedene Sensoren und Aktuatoren, und wird näher auf die Umwandlung zwischen einer analogen, physikalischen Größe und einer digitalen, elektrischen Größe eingehen. Die Studierenden werden hierbei in die Lage versetzt, physikalische Größen anhand ihrer Eigenschaften zu klassifizieren und eine Entscheidung über die für die Umwandlung benötigten Sensoren, Aktuatoren und Hardwarekomponenten zu treffen.</p> <p>Der Lehrstoff wird anhand von Fallstudien z.B. aus den Bereich Automobil exemplifiziert und von den Studierenden anhand eines einfachen eingebetteten Systems angewendet.</p> <p>Schlüsselqualifikationen: Analytisch-methodische Kompetenz, Abwägung von Lösungsansätzen, Präsentation von Lösungen von Übungsaufgaben; Fertigkeit zur verständlichen Darstellung und Dokumentation von Ergebnissen; Fähigkeit, vorhandenes Wissen selbstständig zu erweitern; Qualitätsbewusstsein, Akribie; Selbstreflexion; verantwortliches Handeln vor dem Hintergrund von Unzulänglichkeit und widerstreitenden Interessen</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 150 Std.</p> <p>15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)</p> <p>15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p> <p>60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p> <p>30 Std. Vorlesung (Präsenzstudium)</p> <p>30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Embedded Hardware (Vorlesung)</p> <p>Lehrformen: Vorlesung</p> <p>Sprache: Deutsch / Englisch</p> <p>SWS: 2</p>		

Inhalte:

- Modellierung von Sensoren und Aktuatoren
- Grundlagen in Elektronik
- Hardwarekomponenten von eingebetteten Prozessoren und SoC
- Serielle Schnittstellen
- Sensoren und Eingabegeräte
- Aktuatoren und Ausgabegeräte

Literatur:

- Yifeng Zhu, Embedded Systems with ARM Cortex-M Microcontrollers in Assembly Language and C, E-Man Press LLC, 2017
- Edward Ashford Lee, Sanjit Arunkumar Seshia, Introduction to Embedded Systems - A Cyber-Physical Systems Approach, MIT Press, 2017
- Elecia White, Making Embedded Systems, O'Reilly Media, 2012
- John Catsoulis, Designing Embedded Hardware, O'Reilly Media, 2005
- Rüdiger R. Asche, Embedded Controller: Grundlagen und praktische Umsetzung für industrielle Anwendungen, Springer, 2016
- Hermann Winner, Stephan Hakuli, Felix Lotz, Christina Singer Hrsg., Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort, Springer, 2015
- Bernhard Grimm, Gregor Häberle, Heinz Häberle, u.a., Fachkunde Industrieelektronik und Informationstechnik, Europa Lehrmittel, 2003

Zugeordnete Lehrveranstaltungen:

Embedded Hardware (Vorlesung)

Veranstaltung wird online/digital abgehalten.

Eingebettete Systeme sind Computer Systeme, deren vorrangige Aufgabe es ist die umgebende physikalische und technische Welt zu erfassen bzw. zu beeinflussen. Diese kann beispielsweise durch regelmäßiges Messen physikalischer Größen oder durch die Steuerung bzw. Regelung von technischen Anlagen geschehen. Um diese Aufgaben zu bewerkstelligen benötigen eingebettete Prozessoren diverse Hardwarekomponenten, die es ihnen ermöglicht mit der Außenwelt zu interagieren. Zusätzlich werden Sensoren und Aktuatoren benötigt. Sie übernehmen die Umwandlung von physikalischen Größen in elektrische Signale und umgekehrt. In dieser Vorlesung beschäftigen wir uns sowohl mit den beschriebenen Hardwarekomponenten als auch mit verschiedenen Sensoren und Aktuatoren. Wir lernen wie ein Prozessor mit Hilfe dieser Komponenten mit seiner Umgebung interagieren kann und welchen Einfluss die Komponenten auf die Wandlung von physikalischen Größen zu einem digitalen Wert (und umgekehrt) haben kann.
 ... (weiter siehe Digicampus)

Modulteil: Embedded Hardware (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Embedded Hardware (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Embedded Hardware

Klausur / Prüfungsdauer: 60 Minuten

Modul INF-0371: Approximation Algorithms <i>Approximation Algorithms</i>		5 ECTS/LP
Version 1.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Tobias Mömke		
Lernziele/Kompetenzen: Developing an understanding of central topics in the field of approximation algorithms; acquiring powerful mathematical tools to analyze algorithms; improve the ability to abstract and systematically solve optimization problems. Schlüsselqualifikationen: Ability to build intuitive understanding of mathematical formalisms; ability to identify core properties of optimization problems; deep understanding of powerful mathematical tools		
Arbeitsaufwand: Gesamt: 150 Std. 30 Std. Übung (Präsenzstudium) 30 Std. Vorlesung (Präsenzstudium) 60 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)		
Voraussetzungen: Basic knowledge of Algorithms and Data Structures (e.g., "INF-0111: Informatik 3") and Theoretical Computer Science (e.g., "INF-0110: Einführung in die Theoretische Informatik").		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 4	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Approximation Algorithms (Vorlesung) Lehrformen: Vorlesung Sprache: Englisch SWS: 2		
Inhalte: Given an NP-hard optimization problem, how well can it be approximated in polynomial time? It is exciting and challenging to understand the approximability of fundamental optimization problems. This course mainly focuses on upper bounds, i.e., designing efficient approximation algorithms. In this course, we will study several classes of problems, such as packing problems, network design, and graph problems. We will cover central algorithmic techniques for designing approximation algorithms, including greedy algorithms, dynamic programming, linear and semi-definite programming, and randomization.		
Literatur: <ul style="list-style-type: none"> • David P. Williamson and David B. Shmoys, The Design of Approximation Algorithms, Cambridge University Press. • Vijay V. Vazirani, Approximation Algorithms, Springer. 		
Zugeordnete Lehrveranstaltungen: Approximation Algorithms (Vorlesung) <i>*Veranstaltung wird online/digital abgehalten.*</i> Given an NP-hard optimization problem, how well can it be approximated in polynomial time? It is exciting and challenging to understand the approximability of fundamental optimization problems. This course mainly focuses		

on upper bounds, i.e., designing efficient approximation algorithms. In this course, we will study several classes of problems, such as packing problems, network design, and graph problems. We will cover central algorithmic techniques for designing approximation algorithms, including greedy algorithms, dynamic programming, linear and semi-definite programming, and randomization. This course does not require specific prerequisite, other than basic knowledge in algorithms and in data structures.

Modulteil: Approximation Algorithms (Übung)

Lehrformen: Übung

Sprache: Englisch / Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Approximation Algorithms (Übung)

Veranstaltung wird online/digital abgehalten.

Prüfung

Approximation Algorithms

Klausur / Prüfungsdauer: 90 Minuten

Modul ZCS-2000: Softskills	2 ECTS/LP
Version 1.0.0 (seit SoSe16) Modulverantwortliche/r: Claudia Lange-Hetmann	
Inhalte: Detailbeschreibungen zu Kursen und Anmeldeverfahren befinden sich auf https://www.uni-augsburg.de/de/studium/zusatzqualifikationen/profilbildung/#Anker_skK bzw. im digicampus.	
Lernziele/Kompetenzen: Die Studierenden erwerben in diesem Modul - je nach Kurswahl - entweder kommunikative, soziale oder methodische Fähigkeiten, die unerlässlich für ihre künftige Berufsfähigkeit sind, denn diese fordert eine überzeugende Persönlichkeit des Einzelnen und eine einwandfreie und zielgerichtete Interaktion im Team. Daher wird bei der Auswahl empfohlen, einen Kurs aus einem der drei Kompetenzgebiete zu wählen, die zur Stärkung der eigenen Persönlichkeit sinnvoll und wichtig sind. Zudem bildet die interdisziplinäre Zusammensetzung der Teilnehmer aus unterschiedlichen Fachrichtungen den typischen Wirkungskreis im späteren Arbeitsumfeld ab. Die Studierenden werden - abhängig von der Kursthemenwahl - - neben dem Erwerb der Fertigkeit der verständlichen, sicheren und überzeugenden Darbietung von Ideen, Konzepten und Ergebnissen bzw. dem Verständnis der psychologischen Grundlagen von Dialogen und Verhandlungen dieses Wissen anwenden können, um Interesse, Verständlichkeit und Sympathie zu erzeugen und zielorientiert zu präsentieren bzw. zu argumentieren. - die Kommunikations-, Dialog- und Teamprozesse in Bezug auf Motivation, Effektivität und kennen die Entstehung, Dynamik, Lösung und Prävention von Konflikten verstehen und können Moderationstechniken und ihre Fertigkeit zur Selbstreflexion anwenden, sie beherrschen die Regeln bei der Teamarbeit, bei Besprechungen bis hin zur Führung von Teams oder kennen den Nutzen von gesellschaftlichem Engagement für sich und die Gesellschaft. - grundlegende Konzepte des Projektmanagements (u.a. Entwurf von strategischen Projektstrukturplänen, Analyse der Projektumwelt/-risiken, Projektcontrolling) verstehen und können die Grundlagen der Motivationspsychologie und zentrale Führungstechniken zur Erreichung des Projekterfolgs anwenden. Oder sie können grundlegende Strategien und Methoden für die Entwicklung und Absicherung einer Unternehmensführung anwenden, sie kennen Marketing- u. Vertriebsstrategien, bewerten deren Erfolgsaussichten und haben Kenntnisse in Personal- und Finanzmanagement. Sie verstehen Probleme zu analysieren und können konstruktiv im Team eine Lösung erarbeiten und kommunizieren oder vertiefen Teilaspekte wie u.a. Kreativität, Innovationsfähigkeit mit innovativen Methoden. Besonderer Wert wird - je nach Kurs - auf die Weiterentwicklung der eigenen Präsentations- und Kommunikationsfähigkeit, der Teamkompetenz sowie die Anwendung des Methodenwissens und die Erreichung realistischer Ziele gelegt. Die interdisziplinäre Herangehensweise an eine Problemstellung wird durch die interdisziplinäre Zusammensetzung der Kleingruppen in den Kursen trainiert, durch praktische Übungen in den Kursen gefestigt und durch Selbstreflexion und Feedbackmethoden verinnerlicht.	
Bemerkung: <u>Anmeldungspflicht:</u> Für die Teilnahme an den Kursen ist eine Anmeldung über digicampus erforderlich. <u>Anmeldephase:</u> Jan (für das folgende SS) bzw. Juli (für das folgende WS). Die Kurse finden größtenteils ab März bis letzten Sa* im April (SS) bzw. ab Sep. bis letzten Sa* im Okt. *vor Vorlesungsbeginn statt sowie einige in der Vorlesungszeit (Fr/Sa). Die Kurse haben eine limitierte Teilnehmerzahl pro Semester.	
Arbeitsaufwand: Gesamt: 60 Std. 20 Std. Seminar (Präsenzstudium) 10 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 20 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 10 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)	

Voraussetzungen: keine		ECTS/LP-Bedingungen: aktive Übungsteilnahme im Kurs
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 2.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile**Modulteil: Softskills****Lehrformen:** Kurs**Sprache:** Deutsch / Englisch**SWS:** 2**ECTS/LP:** 2.0**Inhalte:**

Zur Auswahl stehen nachfolgende Kurse/Themen:

(1) Kommunikationskompetenz

- Kommunikationstraining
- Rhetorik
- Präsentation (1 bz. 2-tägige Kurse)
- Strategische Gesprächsführung
- Feedback geben (1 Tag - daher 2. Kurs notwendig!)
- Kommunikation in Projekten (1 Tag - daher 2. Kurs notwendig!)
- Communication in engl. (1 Tag - daher 2. Kurs notwendig!)

(2) Sozialkompetenz

- Konfliktmanagement
- Emotionale Intelligenz
- Moderation & Teamleitung
- Führungskompetenzen entwickeln
- Gesellschaftliches Engagement

(3) Methodenkompetenz

- Zeit-/Selbstmanagement
- Changemanagement
- Meetings erfolgreich moderieren (1 Tag - daher 2. Kurs notwendig!)
- Innovationen entwickeln
- Design Thinking
- Projektmanagement (dt./ engl.)
- Unternehmerisches Denken

Weiterehin können auch Kompakt-Kurse gewählt werden, bei denen die Teilnehmer o.g. Fähigkeiten erlernen und eine Projektaufgabe im Team bearbeiten. Der höhere Zeitaufwand wird mit mehr Erfahrung honoriert.

- Kompaktkurs Märkte für Menschen
- Kompaktkurs Projekte real durchführen
- Kompaktkurs Projekt "Expedition"
- ACE Startup-Challenge

Detailbeschreibungen zu allen Kursen sowie die konkreten Kursthemen und Termine pro Semester im digicampus.

Lehr-/Lernmethoden:

Vortrag, Diskussion, Übungen, Praxisbeispiele, event. Projektarbeit unter Verwendung von multimedialen Techniken (Beamer, Flipchart, Pinwand)

Literatur:

wird im Kurs bz. in die Kursbeschreibungen angeben bzw. vorab kommuniziert.

Zugeordnete Lehrveranstaltungen:

Kompaktkurs - Projekt "Expedition" (Kurs)

Veranstaltung wird in Präsenz abgehalten.

Projektarbeit wird sowohl im Studium als auch im Beruf gefordert und verlangt neben fachlichem und methodischem Knowhow auch Fähigkeiten wie Kommunikationsgeschick und Verantwortlichkeitsgefühl. Lernen Sie Projekte effizient und geordnet sowie mit Freude durchzuführen, die Teammitglieder zu motivieren und nach ihren Fähigkeiten einzusetzen, gemeinsam auf ein sinnvolles Ziel zuzusteuern und am Ende das Ergebnis gemeinsam entsprechend in Szene zu setzen. In diesem Intensivkurs, werden Sie mit fachlicher Anleitung ein mehrtägiges Projekt aus dem realen Arbeitsalltag des Projektpartners durchführen und dabei wertvolle Erfahrungen sammeln. Begleitend werden Sie Ihre Potentiale und die des Teams entdecken und stärken lernen, denn unser Projektpartner Outward Bound Germany bietet neben der Projektaufgabe auch ein interessantes Kurssetting. Lerninhalte: • Aufbau von Projektteams und die Verschiedenen Rollen /Aufgaben • Projektplanung und zielspezifische Umsetzung • Projektpräsentation und -absc
... (weiter siehe Digicampus)

Kompaktkurs - Projekte real durchführen (Kurs)

Veranstaltung wird online/digital abgehalten.

Projektarbeit und Team-Work brauchen Sie sowohl im Studium als auch im späteren Berufsleben. Gleichmaßen ist es heute im Arbeitsalltag unumgänglich, mit innovativen und agilen Methoden an komplexe Fragestellungen heranzugehen. Dieser Intensivkurs stellt Sie auf die Probe! Arbeiten Sie im Team eine Woche an einem konkreten Projekte. Lernen Sie am Vormittag in intensiven Online-Seminaren ausgewählte Inhalte aus Teamführung, agilem Projektmanagement, Design Thinking, Business Modelling und sicherem Präsentieren. Am Nachmittag arbeiten Sie in Ihrem Team remote an einem Projekt. Zusätzlich erarbeiten Sie sich im Selbststudium auf einer Lernplattform weitere Inhalte. Jeder Tag wird mit einem Review der Ergebnisse beendet. Begleitend werden Sie Ihre Potentiale und die des Teams entdecken und stärken lernen, denn unser Projektpartner Steinbeis IFEM bietet neben der Projektaufgabe eine langjährige Erfahrung bei der Durchführung von Trainings in allen Ebenen diverser Unternehmen und Organisati
... (weiter siehe Digicampus)

Kompaktkurs Märkte für Menschen - Veränderungen gestaltet (Kurs)

Veranstaltung wird als Hybrid/gemischt abgehalten.

Inhalte: In diesem Kompaktkurs werden Inhalte von gesellschaftlicher Relevanz interdisziplinär erschlossen. Hierbei werden Themenbereiche wie Ernährungsgewohnheiten vs. globale Ernährungssicherheit, (Finanz-)Märkte vs. Gerechtigkeit kontrovers diskutiert. Jeder dieser Themenbereiche wird zunächst mit Hilfe wissenschaftlicher Inputs interdisziplinär erschlossen. Daran anschließend erarbeitet sich jede Kleingruppe fundiertes Wissen einem Themenbereich, setzt sich differenziert mit kontroversen Perspektiven auseinander und präsentiert die Ergebnisse anschließend kontrovers in einer „Talk-Show“ (Podiumsdiskussion). Die Studierenden erarbeiten in Projektgruppen selbst konkrete Vorschläge und Projekte zur Gestaltung einer besseren Zukunft im Rahmen der Themenkomplexe und präsentieren diese zum Abschluss des Seminars. Parallel zu dieser inhaltlichen Arbeit werden methodische Kenntnisse und Hintergrundinformationen vermittelt, um die TeilnehmerInnen dazu zu befähigen, selbst die Zukunft aktiv
... (weiter siehe Digicampus)

Kompaktkurs: ACE Startup Challenge (Kurs)

Veranstaltung wird online/digital abgehalten.

Im innovativen, interdisziplinaren Seminarkonzept bekommen die Studierenden einen Startup Real-Case, an dem Sie ihr ganzes unternehmerisches Talent unter Beweis stellen dürfen. Dabei werden sie von Experten und Startup-Mentoren begleitet; Fachvorträge und die Möglichkeit mit der Augsburger Startup Szene auf Tuchfühlung zu gehen, runden das Seminarangebot ab. Nach der Teilnahme der Modulveranstaltung sind die Studierenden in der Lage, wesentliche Themen- und Problemstellungen rund um Startups und Unternehmensgründungen zu kennen, zu analysieren und geeignete Strategie abzuleiten. Insbesondere erhalten die Studierenden ein tiefes Verständnis für Geschäftsmodelle, Fragen des Pricings und Marketings sowie rechtliche Fragestellungen. Die Studierenden erlernen Kenntnisse zur Anwendung der wichtigsten Methoden und Konzepte zum Business Modeling, der

Budgetierung, Marktanalysen und Unternehmensfinanzierung. Die Studierenden können nach dem Studium des Moduls unternehmerische Problemstellungen
... (weiter siehe Digicampus)

Kurs Business Knigge erleben (Kurs)

Veranstaltung wird online/digital abgehalten.

Warum ist es auch heute noch wichtig, angemessene Umgangsformen und ein passendes Auftreten bei öffentlichen Anlässen zu kennen und anzuwenden? Sind „Knigge und Etikette“ noch aktuell? In Theorie und Praxis werden wir Themen wie passende Begrüßungsformen und Verabschiedungen, Small Talk, Umgangsformen, authentisches Auftreten und vieles mehr gemeinsam bearbeiten. Wir werden gemeinsam diskutieren, spielerisch die Fragestellungen erarbeiten und uns austauschen. Gezeit werden Sie angemessene Umgangsformen im Beruf und Gesellschaft kennen lernen und ein Bewusstsein entwickeln für persönliche Entscheidung, um beim professionellen Auftreten authentisch zu bleiben. Kursinhalte: o Begrüßung und Empfang – der erste Eindruck zählt o Verbale und non-verbale Kommunikation – small talk und mehr o Kleider machen Leute – wie kleide ich mich richtig im Beruf? o Mein persönlicher Stil – was ist authentisch und passt zu mir? o Persönlicher Kontakt - Umgangsformen (am Tisch) o Professioneller Kontakt pe
... (weiter siehe Digicampus)

Kurs Changemanagement (Kurs)

Veranstaltung wird online/digital abgehalten.

Veränderungen effizient gestalten, Widerstände positiv wandeln. Als Fach- und Führungskraft ist es Ihre Aufgabe, Veränderungen in Ihrem Unternehmen aktiv zu gestalten und erfolgreich umzusetzen. Wie können Unternehmen die Herausforderungen einer sich ständig wandelnden Welt begegnen um ihr Überleben zu sichern? Möchten Sie Veränderungen erfolgreich zum Ziel bringen und Ursachen von Widerständen verstehen? Wollen Sie wissen, wie Sie in schwierigen Situationen schneller Lösungen finden und ein Team firmieren, das effizient zusammenarbeitet? Change Management kann Ihnen dabei helfen, den notwendigen Wandel systematisch, dh. bewusst zu gestalten. Veränderungen gehen dabei oft mit Ängsten und einer Abwehrhaltung der Menschen einher. Lernen Sie in diesem Kurs Veränderungen erfolgreich zu bewältigen und mit Widerständen umzugehen. Erfahren Sie sehr anschaulich, wie Veränderungsprozesse gesteuert werden können, Widerstände erst gar nicht entstehen und falls doch zielorientiert aufgelöst werden
... (weiter siehe Digicampus)

Kurs Emotionale Intelligenz (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Die Forschung zur Emotionalen Intelligenz zeigt, dass Schlüsselkompetenzen neben fachlichen und funktionalen Kompetenzen für den Studienerfolg und die berufliche Karriere von ausschlaggebender Bedeutung sind (Goleman, 2017). Der Kurs „Emotionale Intelligenz“ baut auf dem gleichnamigen Konzept von Daniel Goleman (2011) auf und gibt einen allgemeinen Überblick über seine Arbeiten und Erkenntnisse. Nach Goleman sind fünf Kompetenzen für den Schul-, Studien- und Berufserfolg sowie generell für eine erfolgreiche Lebensführung verantwortlich: Selbstreflexion, Selbstbeherrschung, Selbstmotivation, Empathie und Soziale Kompetenz. Im Kurs werden diese erfolgsrelevanten Kompetenzen durch geeignete Methoden und didaktische Ansätze bei den Kursteilnehmern konsequent (weiter) entwickelt. Auf diese Weise erwerben die Teilnehmer nicht nur relevantes Wissen über die Emotionale Intelligenz, sondern auch praktische Fähigkeiten zu ihrer wirksamen Anwendung. Weitere zentrale Inhalte des Kurses sind: - Die
... (weiter siehe Digicampus)

Kurs Emotionale Intelligenz (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Die Forschung zur Emotionalen Intelligenz zeigt, dass Schlüsselkompetenzen neben fachlichen und funktionalen Kompetenzen für den Studienerfolg und die berufliche Karriere von ausschlaggebender Bedeutung sind (Goleman, 2017). Der Kurs „Emotionale Intelligenz“ baut auf dem gleichnamigen Konzept von Daniel Goleman (2011) auf und gibt einen allgemeinen Überblick über seine Arbeiten und Erkenntnisse. Nach Goleman sind fünf Kompetenzen für den Schul-, Studien- und Berufserfolg sowie generell für eine erfolgreiche Lebensführung verantwortlich: Selbstreflexion, Selbstbeherrschung, Selbstmotivation, Empathie und Soziale Kompetenz. Im Kurs werden diese erfolgsrelevanten Kompetenzen durch geeignete Methoden und didaktische Ansätze bei den

Kursteilnehmern konsequent (weiter) entwickelt. Auf diese Weise erwerben die Teilnehmer nicht nur relevantes Wissen über die Emotionale Intelligenz, sondern auch praktische Fähigkeiten zu ihrer wirksamen Anwendung. Weitere zentrale Inhalte des Kurses sind: - Die ... (weiter siehe Digicampus)

Kurs Emotionale Intelligenz (Option 3) (Kurs)

Veranstaltung wird online/digital abgehalten.

Die Forschung zur Emotionalen Intelligenz zeigt, dass Schlüsselkompetenzen neben fachlichen und funktionalen Kompetenzen für den Studienerfolg und die berufliche Karriere von ausschlaggebender Bedeutung sind (Goleman, 2017). Der Kurs „Emotionale Intelligenz“ baut auf dem gleichnamigen Konzept von Daniel Goleman (2011) auf und gibt einen allgemeinen Überblick über seine Arbeiten und Erkenntnisse. Nach Goleman sind fünf Kompetenzen für den Schul-, Studien- und Berufserfolg sowie generell für eine erfolgreiche Lebensführung verantwortlich: Selbstreflexion, Selbstbeherrschung, Selbstmotivation, Empathie und Soziale Kompetenz. Im Kurs werden diese erfolgsrelevanten Kompetenzen durch geeignete Methoden und didaktische Ansätze bei den Kursteilnehmern konsequent (weiter) entwickelt. Auf diese Weise erwerben die Teilnehmer nicht nur relevantes Wissen über die Emotionale Intelligenz, sondern auch praktische Fähigkeiten zu ihrer wirksamen Anwendung. Weitere zentrale Inhalte des Kurses sind: - Die ... (weiter siehe Digicampus)

Kurs Innovationen entwickeln (Kurs)

Veranstaltung wird online/digital abgehalten.

Großartige Innovationen wie das iPhone, die VR-Brille Oculus Rift oder der Tesla Model S sind der Traum eines jeden Unternehmers. Doch wie kommt man auf solche Ideen und wie wird daraus ein erfolgreiches Produkt? Wie das Scheitern des Alpha Patentfonds zeigt, sind Ideen und selbst erteilte Patente nicht zwingend durchstartend. Eine Idee wird dann zur Innovation, wenn es gelingt, sie zur Realisierungsreife zu bringen und Vorgesetzte, Partner und Geldgeber davon zu begeistern, die Marktchancen realistisch eingeschätzt werden, die technische Machbarkeit und vor allem der wirtschaftliche Nutzen gegeben sind sowie das Projekt- oder Gründerteam und das richtige Timing im Marktumfeld stimmt. Kursinhalte: • Innovationsentwicklung und –identifikation mithilfe von - Kreativitätstechniken (Brainstorming, Rollenwechsel, Bionik, Was-wäre-wenn-Methode, Denkhüte von De Bono) - Trendadaption (360° Scanning, Trendanalyse) - Design Thinking Methode • Praktische Anwendung als Innovationsprozess in der Gr ... (weiter siehe Digicampus)

Kurs Kommunikationstraining (Kurs)

Veranstaltung wird online/digital abgehalten.

In diesem Online-Seminar lernen Sie durch authentische wertschätzende Kommunikation gerade in herausfordernden Zeiten zu begeistern, Emotionen zu wecken und erfolgreich einzusetzen. Erleben Sie, wie Sie professionell strukturierte Gespräche effektiv, klar und überzeugend führen. Die Inhalte des praxisorientierten Online Trainings sind sofort in Ihrem Studium und Alltag erfolgreich einsetzbar! Inhalte: • Kommunikation gerade in schwierigen Zeiten zielorientiert ausrichten • Überzeugende Fragetechnik – Wer fragt, der führt • In 60 Sekunden begeistern mit strukturierter Elevator Pitch • Menschen- und Kundentypen besser einschätzen • Gespräche positiv mit Anerkennung und Wertschätzung steuern • Tipps und Werkzeuge für mehr Selbstsicherheit in der Kommunikation Methodik: Aktives praxisorientiertes Live Online Training mit Partnerübungen im virtuellen Klassenraum und Einzelübungen. Leistungsnachweis: Anwesenheit an allen synchronen Terminen, Bearbeitung aller Arbeitsaufträge. Dozentin: Micha ... (weiter siehe Digicampus)

Kurs Konfliktmanagement (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Konflikte gehören zum Alltag wie auch zum Berufsleben. Konflikte sind allgegenwärtig. Umso wichtiger ist es zu wissen, wie man Konflikte konstruktiv löst und für beide Seiten gewinnbringend in Positives verwandeln kann. Lernen Sie sich und Ihre Mitmenschen besser kennen. Wir erarbeiten mit Ihnen zusammen die Techniken, um auch in schwierigen Situationen gelassen und zielorientiert zu agieren. Lerninhalte: • Was ist ein Konflikt? • Wie entsteht er? • Wie löst man Konflikte konstruktiv? • Nullsummenspiel vs. Win-Win Situation • Killerphrasen

entlarven Methoden: Theorie in Kombination mit konkreten Beispielen aus der Praxis, praktische Übungen und viele unterschiedliche Tools Dozentin: Natalie Raess-Beuchle, Coraessco Coaching & Consulting

Kurs Konfliktmanagement (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Konflikte sind allgegenwärtig. Sie gehören sowohl zum privaten Alltag wie auch zum Berufsleben. Daher ist es wichtig zu wissen, wie man Konflikte konstruktiv löst und für alle Beteiligten gewinnbringend in positive Veränderungen verwandeln kann. Die Teilnehmer erarbeiten gemeinsam Vorgehensweisen, um auch in schwierigen Konfliktsituationen gelassen und zielorientiert zu agieren. Vor dem Hintergrund der Prinzipien gewaltfreier Kommunikation (Rosenberg, 2016) lernen die Teilnehmer, welche Gesprächshaltungen, -formen und -techniken zur Konflikteskalation bzw. zur De-Eskalation beitragen. Mittels der gewaltfreien Bewegungskunst „Aikido“ können die Teilnehmer in Körperübungen erleben, wie man Angriffsenergien aufnehmen und umlenken kann. Aikido zeigt, wie man Konflikte ohne Konfrontation oder Widerstand behandelt. Anschließend wird der Bewegungsablauf des Aikido durch Analogiebildung auf das Führen von Konfliktgesprächen übertragen. Das entsprechende Gesprächsmodell ist Aikikom (= Kommunika
... (weiter siehe Digicampus)

Kurs Meetings erfolgreich moderieren (Kurs)

Veranstaltung wird online/digital abgehalten.

Eventuell bereits im Studium und sicher im Berufsleben sind Besprechungen ständige Begleiter. Bestens vorbereitete und erfolgreich durchgeführte Besprechungen sind dennoch eine Seltenheit. Dabei kann man gutes Besprechungsmanagement ganz einfach trainieren und mit dieser Kompetenz in Zukunft glänzen. Inhalte:
• Welche Besprechungsarten gibt es? • Wie bereite ich eine Besprechung professionell vor? • Wie leite ich zielführend durch die verschiedenen Besprechungsphasen? • Wie bringe ich meine Botschaft überzeugend und zielgruppengerecht an den Mann/die Frau? • Wie nutze ich dabei Visualisierungen? • Wie bringe ich Besprechungen zu einem verbindlichen Abschluss? • Wie gehe ich mit unterschiedlichen Besprechungssituationen um? Methoden: Theorie-Input, Tipps aus der Praxis, Fallbeispiele und Übungen, Reflexion, ausführliches Feedback, Kollegiale Beratung, Diskussion und Einzel-Coaching. Dozentin: Nina Turani, Seniorreferentin Personal- und Führungskräfteentwicklung, DB Fernverkehr AG
... (weiter siehe Digicampus)

Kurs Project management (english) (Kurs)

Veranstaltung wird online/digital abgehalten.

Projects are important at all company aspects and resorts. Essential for success is that all project members know and accept the project goals, plan and their own tasks as well as an efficient project coordination and controlling. Therefore the course trains fundamental concepts of modern project management. Lerninhalte: • Basics of Project management • Project definition • Project organisation • Project communication • Project planning • Project calculation • Project risk management • Project controlling • Project closing • Project documentation Methoden: teacher presentation (in parts), working on a business case in small groups, presenting the business case solution per team at the end, detailed feedback from all attendencies and course leader Dozent: Prof. Dr. Matthias Menter (Jun.-Prof.), Chair of Business Dynamics, Innovation and Economic Change, Friedrich Schiller University Jena
... (weiter siehe Digicampus)

Kurs Projektmanagement (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Projekte stellen eine immer bedeutsamer werdende Form zur Unternehmensführung dar. Maßgeblich für deren Erfolg sind effiziente Koordinierung sowie zielfördernde Beiträge seitens der Projektbeteiligten. Daher vermittelt dieser Kurs grundlegende Konzepte modernen Projektmanagements. Lerninhalte: • Grundlagen des Projektmanagements • Projekte auswählen und Projektziele definieren • Projekte planen und effizient kontrollieren • Projektstrukturpläne entwerfen und Meilensteine setzen • Projekte kosteneffizient kalkulieren • Projektrisiken erkennen und managen • Projekte zielorientiert dokumentieren • Projekte erfolgreich abschließen Methoden: Vortrag durch Referenten, Fortlaufende Bearbeitung einer Fallstudie in Kleingruppen, Abschlusspräsentation der jeweiligen Fallstudie durch die Kursteilnehmer, Ausführliches Feedback durch Kursteilnehmer und -leiter Dozent: Prof. Dr. Matthias Menter (Jun.-Prof.), Lehrstuhl für Unternehmensentwicklung, Innovation und wirtschaftlichen Wandel, Friedrich-Schill

... (weiter siehe Digicampus)

Kurs Projektmanagement (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Projekte stellen eine immer bedeutsamer werdende Form zur Unternehmensführung dar. Maßgeblich für deren Erfolg sind effiziente Koordinierung sowie zielfördernde Beiträge seitens der Projektbeteiligten. Daher vermittelt dieser Kurs grundlegende Konzepte modernen Projektmanagements. Lerninhalte: • Grundlagen des Projektmanagements • Projekte auswählen und Projektziele definieren • Projekte planen und effizient kontrollieren • Projektstrukturpläne entwerfen und Meilensteine setzen • Projekte kosteneffizient kalkulieren • Projektrisiken erkennen und managen • Projekte zielorientiert dokumentieren • Projekte erfolgreich abschließen Methoden: Vortrag durch Referenten, Fortlaufende Bearbeitung einer Fallstudie in Kleingruppen, Abschlusspräsentation der jeweiligen Fallstudie durch die Kursteilnehmer, Ausführliches Feedback durch Kursteilnehmer und -leiter Dozent: Prof. Dr. Matthias Menter (Jun.-Prof.), Lehrstuhl für Unternehmensentwicklung, Innovation und wirtschaftlichen Wandel, Friedrich-Schill

... (weiter siehe Digicampus)

Kurs Projektmanagement (Option 3) (Kurs)

Veranstaltung wird online/digital abgehalten.

Projekte stellen eine immer bedeutsamer werdende Form zur Unternehmensführung dar. Maßgeblich für deren Erfolg sind effiziente Koordinierung sowie zielfördernde Beiträge seitens der Projektbeteiligten. Daher vermittelt dieser Kurs grundlegende Konzepte modernen Projektmanagements. Lerninhalte: • Projektanforderungen definieren & Mitarbeiter für sich gewinnen • Entwerfen von strategischen Projektstrukturplänen • Analyse von Projektumwelt und –risiken • Umgehen von Fallstricken bei verteilten Teams • Die fünf wichtigsten Führungstechniken • Projekt- und Fortschrittscontrolling – immer alles im Griff • Sieben Erfolgsstrategien für höhere Motivation Methoden: Fortlaufende Bearbeitung einer Fallstudie in Kleingruppen, Abschlusspräsentation der jeweiligen Fallstudie durch die Kursteilnehmer, ausführliches Feedback durch Kursteilnehmer und -leiter Dozentin: Sabine Schumann, Trainerin und Projektmanagement (GPM)

... (weiter siehe Digicampus)

Kurs Präsentationstraining (Kurs)

Veranstaltung wird online/digital abgehalten.

Wirkungsvoll präsentieren: Menschen begeistern – Emotionen erzeugen – Erfolg steigern Dieser Online-Kurs vermittelt Ihnen das nötige Know-how und gibt viele praxiserprobte Werkzeuge und Tipps an die Hand, um souverän, wirkungsvoll und begeisternd zu moderieren und zielführend zu präsentieren. Die Inhalte des praxisorientierten Präsenz-Online Trainings sind sofort in Ihrem Alltag und Studium erfolgreich einsetzbar! Inhalte: • Ablauf einer Präsentation effektiv vorbereiten und gestalten • Die 10 Module einer erfolgreichen Präsentation • Professionelle Vorstellung und Begrüßung • Souverän Auftreten und Begeisterung ausstrahlen • Körpersprache, Sprache und Stimme wirkungsvoll einsetzen • Die Zuhörer für sich gewinnen, motivieren und Interesse wecken • Tipps und Tricks zur Steigerung der Publikumsaufmerksamkeit Methodik: Aktives praxisorientiertes Online Training mit Partnerübungen im virtuellen Klassenraum und Einzelübungen. Leistungsnachweis: Anwesenheit an allen synchronen Terminen, Bear

... (weiter siehe Digicampus)

Kurs Rhetorik (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Den Zuhörer in den Bann ziehen – in Bildern sprechen. Lernen Sie die Kunst des Sprechens sowie Gedächtnisstützen, damit Sie überzeugend und frei vortragen können. Dieses besondere Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Rede. In Zukunft wird Ihre Stimme süchtig machen. Überzeugen Sie ab heute jeden durch unschlagbare Argumentationsketten. Lerninhalte: • Körpersprache – der perfekte Auftritt • Von der Idee zum fesselnden Vortrag – die optimale Herangehensweise & Umsetzung • Packende Argumentationsstrategien – Überrede nicht, überzeuge! • Wirksprache – wirken, nachwirken, im Gedächtnis bleiben. • Sprachkompetenz und Emotionale Intelligenz – damit du einen rundherum positiven Eindruck hinterlässt • Merk- & Präsentationstechniken – dein Leitfaden, um mit wenig alles vorzutragen • Deine Stimme, dein Instrument – Trainingstipps für den richtigen Ton • Sprachkompetenz – Worte, meine Fallschirme... wer euch richtig öffnet, schwebt! • Rhetorik im Alltag – B

... (weiter siehe Digicampus)

Kurs Rhetorik (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Den Zuhörer in den Bann ziehen – in Bildern sprechen. Lernen Sie die Kunst des Sprechens sowie Gedächtnisstützen, damit Sie überzeugend und frei vortragen können. Dieses besondere Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Rede. In Zukunft wird Ihre Stimme süchtig machen. Überzeugen Sie ab heute jeden durch unschlagbare Argumentationsketten. Lerninhalte: • Körpersprache – der perfekte Auftritt • Von der Idee zum fesselnden Vortrag – die optimale Herangehensweise & Umsetzung • Packende Argumentationsstrategien – Überrede nicht, überzeuge! • Wirksprache – wirken, nachwirken, im Gedächtnis bleiben. • Sprachkompetenz und Emotionale Intelligenz – damit du einen rundherum positiven Eindruck hinterlässt • Merk- & Präsentationstechniken – dein Leitfaden, um mit wenig alles vorzutragen • Deine Stimme, dein Instrument – Trainingstipps für den richtigen Ton • Sprachkompetenz – Worte, meine Fallschirme... wer euch richtig öffnet, schwebt! • Rhetorik im Alltag – B

... (weiter siehe Digicampus)

Kurs Rhetorik (Option 3) (Kurs)

Veranstaltung wird online/digital abgehalten.

Den Zuhörer in den Bann ziehen – in Bildern sprechen. Lernen Sie die Kunst des Sprechens sowie Gedächtnisstützen, damit Sie überzeugend und frei vortragen können. Dieses besondere Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Rede. In Zukunft wird Ihre Stimme süchtig machen. Überzeugen Sie ab heute jeden durch unschlagbare Argumentationsketten. Lerninhalte: • Körpersprache – der perfekte Auftritt • Von der Idee zum fesselnden Vortrag – die optimale Herangehensweise & Umsetzung • Packende Argumentationsstrategien – Überrede nicht, überzeuge! • Wirksprache – wirken, nachwirken, im Gedächtnis bleiben. • Sprachkompetenz und Emotionale Intelligenz – damit du einen rundherum positiven Eindruck hinterlässt • Merk- & Präsentationstechniken – dein Leitfaden, um mit wenig alles vorzutragen • Deine Stimme, dein Instrument – Trainingstipps für den richtigen Ton • Sprachkompetenz – Worte, meine Fallschirme... wer euch richtig öffnet, schwebt! • Rhetorik im Alltag – B

... (weiter siehe Digicampus)

Kurs Selbstmanagement & Produktivitätssysteme (Kurs)

Veranstaltung wird online/digital abgehalten.

Das Studium überdurchschnittlich gut abschließen und trotzdem noch genügend Freizeit haben? In diesem Power-Workshop werden Ihnen Instrumente zur Produktivitäts- und Effektivitätssteigerung an die Hand gegebene und vermittelt, wie Sie sich am besten selbst organisieren, um konstant Fortschritt in Ihren Projekten zu erzielen. Es wird ein im Businesskontext erprobtes Produktivitätssystem vorgestellt, das jeder Teilnehmer individuell auf seine Bedürfnisse zuschneiden kann. Am ersten (halben) Tag des Webinars werden aktuelle Methoden des Selbstmanagements und der Produktivitätsforschung kompakt vorgestellt und miteinander verknüpft. Unter Anleitung des Dozenten definieren Sie Ihre Ziele und leiten umsetzbare Projekte ab. Am zweiten (halben) Tag werden Sie ein digitales Produktivitätssystem aufbauen, das Sie bei der Erreichung der gesetzten Ziele unterstützen wird. Ein digitales Produktivitätssystem ist ein Zusammenschluss von verschiedenen Programmen, das den Anwender dabei unterstützen so

... (weiter siehe Digicampus)

Kurs Strategische Gesprächsführung (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

• Sie wollen die nächste Verhandlung für sich entscheiden? • Lernen Sie konstruktive Verhandlungstaktiken und -strategien kennen. • Erfahren Sie was, es bedeutet überzeugend zu agieren und gelungene Verhandlungen zu führen. Wir verhandeln zu jeder Zeit, nur ist es uns oftmals nicht bewusst. Dieses außergewöhnliche Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Verhandlung. Sie lernen konstruktive Verhandlungstaktiken und -strategien kennen und erfahren, wie Sie durch den Einsatz von gezielten Verhandlungstechniken wesentliche Vorteile für sich nutzen können. Sie werden zukünftig den Verhandlungspartner besser einschätzen, seine Verhandlungsstrategien erkennen und Ihre eigene Position in Verhandlungen überzeugender und nutzenorientierter darstellen können. Lerninhalte: • Psychologische

Grundlagen effektiv nutzen • Sympathie im Gespräch erzeugen • Die Basics aus der Kunst der Diplomatie & die goldenen Regeln der Gesprächsführung • Den Mittelpunkt geschic
... (weiter siehe Digicampus)

Kurs Strategische Gesprächsführung (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Oft setzen sich nicht die besten Ideen durch, sondern die, die mit dem größten Nachdruck vertreten werden. Blieben Ihre Ideen auf der Strecke? Stecken Sie in Diskussionen oft zurück, um andere nicht vor den Kopf zu stoßen und ärgern sich danach darüber? In diesem Seminar trainieren Sie das, was Ihnen wichtig ist, auch gegen Widerstände vertreten und wenn möglich durchsetzen. Sie lernen selbstbewusst für Ihre Ziele und Interessen einzustehen, ohne die Beziehung zu Ihren Gesprächspartnern auf eine Zerreißprobe zu stellen. In Rollenspielen, die an den Berufsalltag angelehnt sind, stärken Sie Ihre Persönlichkeit, indem Sie üben erfolgreich zu diskutieren, zu debattieren und auch mal selbstbewusst „Nein“ zu sagen. Das Training vermittelt Ansätze und Techniken, um in Verhandlungen nachhaltig besser und erfolgreicher abzuschneiden. Sie lernen, auch mit schwierigen Verhandlungspartnern und -Situationen umzugehen. Lerninhalte: • Bedeutung der inneren Einstellung für Selbstbewusstsein, Selbstsic
... (weiter siehe Digicampus)

Kurs Strategische Gesprächsführung (Option 3) (Kurs)

Veranstaltung wird online/digital abgehalten.

• Sie wollen die nächste Verhandlung für sich entscheiden? • Lernen Sie konstruktive Verhandlungstaktiken und -strategien kennen. • Erfahren Sie was, es bedeutet überzeugend zu agieren und gelungene Verhandlungen zu führen. Wir verhandeln zu jeder Zeit, nur ist es uns oftmals nicht bewusst. Dieses außergewöhnliche Seminar erklärt Ihnen praxisnah die wichtigsten Erfolgsregeln für eine gelungene Verhandlung. Sie lernen konstruktive Verhandlungstaktiken und -strategien kennen und erfahren, wie Sie durch den Einsatz von gezielten Verhandlungstechniken wesentliche Vorteile für sich nutzen können. Sie werden zukünftig den Verhandlungspartner besser einschätzen, seine Verhandlungsstrategien erkennen und Ihre eigene Position in Verhandlungen überzeugender und nutzenorientierter darstellen können. Lerninhalte: • Psychologische Grundlagen effektiv nutzen • Sympathie im Gespräch erzeugen • Die Basics aus der Kunst der Diplomatie & die goldenen Regeln der Gesprächsführung • Den Mittelpunkt geschic
... (weiter siehe Digicampus)

Kurs Teamleitung (Option 1) (Kurs)

Veranstaltung wird online/digital abgehalten.

Ziel des Seminars ist es, die Herausforderungen und Potentiale von Teams zu verstehen und nutzen zu lernen. Dafür werden Sie verschiedene Methoden kennenlernen, wie sie Ihr Team für die gemeinsamen Ziele begeistern und dorthin führen können. In interaktiven Übungen werden Sie das neugelernte Wissen vertiefen und Sie erleben direkt den Zusammenhalt und das Führungsverhalten von anderen und sich selbst. Inhalte: • Rhetorik – Ihre Gruppe für Ihre Ideen begeistern • Methoden der Moderation – Die besten Tricks, wie sie eine Gruppe moderieren und dynamische Arbeitsprozesse entstehen lassen. • Führungsstile – Entdecken Sie Ihren persönlichen Führungsstil • Konflikt- & Stressmanagement – Konflikte innerhalb des Teams vermeiden und gemeinsam entspannt ans Ziel • Zielsetzung – Wie Sie Ziele in einem Gruppenprojekt definieren Methoden: Theorie-Input, Tipps aus der Praxis, Fallbeispiele und Übungen, Reflexion, ausführliches Feedback, Diskussion
Dozenten: Max Daufrathhofer, Dr. Philipp Rodrian - be
... (weiter siehe Digicampus)

Kurs Teamleitung (Option 2) (Kurs)

Veranstaltung wird online/digital abgehalten.

Ziel des Seminars ist es, die Herausforderungen und Potentiale von Teams zu verstehen und nutzen zu lernen. Dafür werden Sie verschiedene Methoden kennenlernen, wie sie Ihr Team für die gemeinsamen Ziele begeistern und dorthin führen können. In interaktiven Übungen werden Sie das neugelernte Wissen vertiefen und Sie erleben direkt den Zusammenhalt und das Führungsverhalten von anderen und sich selbst. Inhalte: • Rhetorik – Ihre Gruppe für Ihre Ideen begeistern • Methoden der Moderation – Die besten Tricks, wie sie eine Gruppe moderieren und dynamische Arbeitsprozesse entstehen lassen. • Führungsstile – Entdecken Sie Ihren persönlichen Führungsstil • Konflikt- & Stressmanagement – Konflikte innerhalb des Teams vermeiden und

gemeinsam entspannt ans Ziel • Zielsetzung – Wie Sie Ziele in einem Gruppenprojekt definieren Methoden: Theorie-Input, Tipps aus der Praxis, Fallbeispiele und Übungen, Reflexion, ausführliches Feedback, Diskussion
Dozent: Andreas Renner / Max Daufratshofer, Steinbeis
... (weiter siehe Digicampus)

Kurs Unternehmerisches Denken - Lean Startup (Kurs)

Veranstaltung wird online/digital abgehalten.

Fach- und Führungskräfte mit technischem, naturwissenschaftlichem, juristischem oder geistes- und sozialwissenschaftlichem Hintergrund werden in ihrem Arbeitsalltag zunehmend mit betriebswirtschaftlichen Fragen konfrontiert oder denken gar über eine Unternehmensgründung nach. In diesem Seminar lernen sie die ökonomischen Grundlagen sowie die entsprechenden Fachbegriffe kennen und können diese sofort im Rahmen einer Unternehmensgründung kompetent anwenden und praxisnah erleben. Teilnehmern mit wenig fundierten bzw. ohne betriebswirtschaftliche Vorkenntnisse bietet die Projektarbeit und das Durchspielen einer konkreten Gründung anhand der Lean Startup Methode und des Business Model Canvas einen interessanten Einstieg in ökonomische Zusammenhänge und betriebswirtschaftliche Entscheidungsparameter. Das Verständnis für unternehmerische Vorhaben sowie der sog. Unternehmergeist kann so bei Teilnehmern unterschiedlicher Zielgruppen in Teams gefördert werden. Lerninhalte: • Interaktives Gruppen
... (weiter siehe Digicampus)

Startup Challenge - Augsburg Center for Entrepreneurship (Projektstudium)

Veranstaltung wird online/digital abgehalten.

• Innovatives Entrepreneurship Format und Startup Projektstudium • Anhand einer realen unternehmerischen Entscheidungssituation können die Studierenden ihr ganzes unternehmerisches Talent testen • Die Studierenden lernen wesentliche Methoden und Konzepte zur Steuerung und Bewertung von Geschäftsmodellen, Pricing Strategien, Vertrieb und Marketing. • Experten und renommierte Startup-Mentoren begleiten die Studierenden in fachlichen Fragen. • Fachvorträge zu ausgewählten Themenstellungen runden das Seminarangebot ab; sowie die Möglichkeit mit der Augsburger Startup Szene auf Tuchfühlung zu gehen.

Startup Challenge - Augsburg Center for Entrepreneurship (Seminar) (Seminar)

Veranstaltung wird online/digital abgehalten.

• Innovatives Entrepreneurship Format und Startup Projektseminar • Anhand einer realen unternehmerischen Entscheidungssituation können die Studierenden ihr ganzes unternehmerisches Talent testen • Die Studierenden lernen wesentliche Methoden und Konzepte zur Steuerung und Bewertung von Geschäftsmodellen, Pricing Strategien, Vertrieb und Marketing. • Experten und renommierte Startup-Mentoren begleiten die Studierenden in fachlichen Fragen. • Fachvorträge zu ausgewählten Themenstellungen runden das Seminarangebot ab; sowie die Möglichkeit mit der Augsburger Startup Szene auf Tuchfühlung zu gehen.

Prüfung

Anwesenheit und aktive Übungsteilnahme im Kurs

Beteiligungsnachweis, unbenotet

Modul INF-0058: Seminar Algorithmen und Datenstrukturen für Master <i>Seminar Algorithms and Data Structures for Masters</i>		4 ECTS/LP
Version 1.5.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
<p>Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, sich fachliche Inhalte der Algorithmik eigenständig aus anspruchsvoller wissenschaftlicher Originalliteratur zu erarbeiten und diese kritisch zu bewerten. Sie können das Gelesene in einen größeren Kontext einordnen, eine sinnvolle Themenauswahl treffen und die gewählten fachlichen Inhalte klar und verständlich schriftlich darstellen und mündlich frei vortragen. Sie verstehen es, einen Vortrag gut zu strukturieren, interessant und motivierend zu gestalten und an die Voraussetzungen der Zuhörer und den vorgegebenen zeitlichen Rahmen anzupassen.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken, zur kritischen Stellungnahme und zur überzeugenden fachlichen Argumentation; Fähigkeit zur Literaturrecherche und zum eigenständigen Arbeiten mit englischsprachiger Fachliteratur; Fähigkeit zur Bewertung von Lösungsansätzen, Verfahren und Techniken unter unterschiedlichen Gesichtspunkten; Qualitätsbewusstsein; Akribie; Kommunikationsfähigkeit; Zeitmanagement.</p>		
<p>Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: Empfehlenswert: Gutes Verständnis des Informatik III-Stoffes.		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Seminar Algorithmen und Datenstrukturen		
Lehrformen: Seminar		
Sprache: Deutsch		
SWS: 2		
Inhalte: Aktuelle und klassische Themen aus dem Bereich Algorithmen und Datenstrukturen werden anhand von Originalliteratur behandelt.		
Literatur: Ausgewählte wissenschaftliche Artikel.		
Prüfung		
Schriftliche Ausarbeitung und Vortrag Seminar		

Modul INF-0070: Seminar Organic Computing <i>Seminar Organic Computing</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen:</p> <p>Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet Organic Computing selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten.</p> <p>Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen.</p> <p>Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an.</p> <p>Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikationen: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewusstsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 120 Std.</p> <p>90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p> <p>30 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen:		
keine		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Organic Computing		
Lehrformen: Seminar		
Sprache: Deutsch		
SWS: 2		
Inhalte:		
Die Themen des Seminars werden jedes Jahr neu festgelegt und aktuellen Trends angepasst.		
Literatur:		
Literatur in Abhängigkeit von den aktuellen Themen: wiss. Paper oder Bücher		

Zugeordnete Lehrveranstaltungen:

Seminar über Organic Computing (Seminar)

Veranstaltung wird online/digital abgehalten.

Prüfung

Vortrag und schriftliche Ausarbeitung

Seminar

Modul INF-0071: Seminar Naturalogische Algorithmen und Multiagentensysteme <i>Seminar Nature inspired Algorithms and Multi Agent Systems</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen:</p> <p>Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf den Gebieten der naturinspirierten Algorithmen sowie den Multiagentensystemen selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten.</p> <p>Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen.</p> <p>Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an.</p> <p>Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikationen: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewusstsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 120 Std.</p> <p>30 Std. Seminar (Präsenzstudium)</p> <p>90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Seminar Naturalogische Algorithmen und Multiagentensysteme</p> <p>Lehrformen: Seminar</p> <p>Sprache: Deutsch</p> <p>SWS: 2</p>		

Inhalte:

In dem Seminar sollen sich die Studenten jeweils einem speziellen Thema in Schnittbereich naturanaloge Algorithmen und Multiagentensysteme genauer beschäftigen. Diese Thema kann ein bestimmte Anwendung, z.B. das Swarmoid-Projekt, sein oder auch eine bestimmte Technik, z.B. für Task Allocation betreffen. Sie erstellen einen etwa 30-minütigen Vortrag zum gegebenen individuellen Thema. In einer schriftlichen Ausarbeitung werden die Erkenntnisse zum Thema zusammengefasst.

Literatur:

wird im Seminar bekanntgegeben

Prüfung

Vortrag und schriftliche Ausarbeitung

Seminar

Modul INF-0095: Seminar Multimedia Computing (MA) <i>Seminar Multimedia Computing & Computer Vision (MA)</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien aus dem Gebiet des Multimedia Computings und maschinellen Sehens (z.B. Bildverarbeitung, Videoverarbeitung, maschinelles Sehen/Hören und Lernen, Bild-/ Videosuche) selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz neuer Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen. Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an. Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikationen: Erlernen von Präsentationstechniken, Literaturrecherche; Arbeit mit englischer Fachliteratur; Grundsätze guter wissenschaftlicher Praxis; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten</p>		
<p>Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Seminar Multimedia Computing (MA) Lehrformen: Seminar Sprache: Deutsch Angebotshäufigkeit: jedes Sommersemester SWS: 2</p>		
Inhalte: Das konkrete Thema des Seminars aus dem weitläufigen Gebiet des Multimedia wird jedes Jahr neu festgelegt und an aktuelle Themen angepasst.		
Literatur: aktuelle Forschungsliteratur		
Zugeordnete Lehrveranstaltungen: Seminar über Multimedia und Maschinelles Sehen (Master) (Seminar) <i>*Veranstaltung wird online/digital abgehalten.*</i>		

Prüfung

Vortrag und schriftliche Ausarbeitung

Seminar

Modul INF-0136: Seminar Software- und Systems Engineering (Master) <i>Seminar Software- und Systems Engineering (Master)</i>		4 ECTS/LP
Version 1.1.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen:</p> <p>Nach dem erfolgreichen Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien aus dem Gebiet des Software und Systems Engineering selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten.</p> <p>Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen.</p> <p>Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an.</p> <p>Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Literaturrecherche • Eigenständiges Arbeiten mit auch englischsprachiger Fachliteratur • Analytisch-methodische Kompetenz • Wissenschaftliche Methodik • Grundsätze guter wissenschaftlicher Praxis • Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation • Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und präziser Argumentation • Qualitätsbewußtsein, Akribie • Kommunikationsfähigkeit • Zeitmanagement • Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten 		
<p>Arbeitsaufwand:</p> <p>Gesamt: 120 Std.</p> <p>90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p> <p>30 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	

Moduleile
<p>Moduleil: Seminar Software- und Systems Engineering (Master) Lehrformen: Seminar Sprache: Deutsch / Englisch Angebotshäufigkeit: jedes Sommersemester SWS: 2</p>
<p>Inhalte: Die konkreten Themen des Seminars beschäftigen sich mit aktuellen Themen des Software und Systems Engineering auf Masterniveau und werden jedes Jahr neu festgelegt und an neue Entwicklungen angepasst.</p>
<p>Literatur: Abhängig von den konkreten Themen des Seminars</p>
<p>Zugeordnete Lehrveranstaltungen:</p> <p>Seminar zu IT-Sicherheit (Seminar) <i>*Veranstaltung wird online/digital abgehalten.*</i> Es werden maximal die 10 unten aufgeführten Themen nach dem Prinzip First Come, First Served vergeben. Um am Seminar teilzunehmen, schicken Sie eine Mail mit ZWEI Themenwünschen an philip.lenzen@informatik.uni-augsburg.de. Folgende Seminarthemen werden angeboten: Themenblock 1: Exploits 1. Angriffe auf Signaturen von PDF-Dokumenten Das "Portable Document Format" (PDF) ist de-facto Standard für elektronische Dokumente, unter anderem für elektronische Verträge und Rechnungen im Staats- und Finanzwesen. Zur Sicherung der Authentizität und Integrität von PDF-Dokumenten werden Digitale Signaturen eingesetzt. Allerdings ist eine Vielzahl an PDF-Viewern verwundbar hinsichtlich neuartiger Angriffe auf Signaturen, die Forscher der Ruhr-Universität Bochum entdeckt haben. Diese sollen im Rahmen dieser Arbeit näher untersucht werden. 2. Automatische Heap-basierte Exploit-Generierung Typische Ansätze der automatisierten Generierung von Exploits (Befehlsketten, die aufgrund von Speicherkorruption be ... (weiter siehe Digicampus)</p>
<p>Prüfung</p> <p>Seminar Software- und Systems Engineering (Master) Seminar / Prüfungsdauer: 45 Minuten Bearbeitungsfrist: 3 Monate</p>

Modul INF-0164: Seminar Theorie verteilter Systeme A <i>Seminar in Theory of Concurrent Systems A</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen:</p> <p>Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet "Theorie verteilter Systeme" selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen.</p> <p>Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an.</p> <p>Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikationen: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von theoretischen Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewußtsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 120 Std. 30 Std. Seminar (Präsenzstudium) 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Theorie verteilter Systeme A		
Lehrformen: Seminar		
Sprache: Deutsch		
SWS: 2		
Inhalte: Es werden Arbeiten zu verschiedenen Themen aus dem Bereich "Theorie verteilter Systeme" behandelt.		
Literatur: wird jeweils bekanntgegeben		

Prüfung

Schriftliche Ausarbeitung

Seminar

Modul INF-0227: Seminar Datenbanksysteme für Master <i>Seminar Database Systems Master</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe16) Modulverantwortliche/r: Prof. Dr. Peter Michael Fischer		
<p>Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet der Datenbanksysteme selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen. Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an. Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden. Schlüsselqualifikationen: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewußtsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten</p>		
<p>Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)</p>		
<p>Voraussetzungen: Modul Datenbanksysteme (INF-0073) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig (i. d. R. im SoSe)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Seminar Datenbanksysteme für Master Lehrformen: Seminar Sprache: Deutsch / Englisch SWS: 2</p>		
<p>Inhalte: Aktuelle Forschungsbeiträge aus den Bereich "Datenbanken und Informationssysteme".</p>		
<p>Literatur: Aktuelle Forschungsbeiträge</p>		

Prüfung

Vortrag und schriftliche Ausarbeitung

Seminar

Modul INF-0232: Seminar Medical Information Sciences (MA) <i>Seminar Medical Information Sciences (MA)</i>		4 ECTS/LP
Version 1.0.0 (seit WS16/17) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet Medical Information Sciences selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen. Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an. Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikationen: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewußtsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten</p>		
<p>Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Medical Information Sciences (Seminar)		
Lehrformen: Seminar		
Sprache: Deutsch		
SWS: 2		
Inhalte: Dieses Seminar soll die Grundlagen der Medical Information Sciences behandeln. Es sind verschiedene Themen zu bearbeiten die als Grundlage für ein nachfolgendes Praktikum dienen sollen.		
Literatur: Wird in der jeweiligen Kickoff-Veranstaltung vorgestellt.		
Zugeordnete Lehrveranstaltungen:		

Seminar Medical Information Sciences (Master) (Seminar)

Veranstaltung wird online/digital abgehalten.

Bestandteil dieses Seminars sind fortgeschrittene Ansätze und Techniken im Bereich Medical Information Science.

Prüfung

Vortrag und schriftliche Ausarbeitung

Seminar

Modul INF-0240: Seminar Informationssysteme für Master <i>Seminar Information Systems Master</i>		4 ECTS/LP
Version 1.0.0 (seit WS16/17) Modulverantwortliche/r: Prof. Dr. Peter Michael Fischer		
<p>Lernziele/Kompetenzen:</p> <p>Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet der Informationssysteme selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen.</p> <p>Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an.</p> <p>Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikationen: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewußtsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 120 Std. 30 Std. Seminar (Präsenzstudium) 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p>		
<p>Voraussetzungen:</p> <p>Modul Datenbanksysteme (INF-0073) - empfohlen</p>		
Angebotshäufigkeit: unregelmäßig (i. d. R. im WS)	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Informationssysteme für Master		
Lehrformen: Seminar		
Sprache: Deutsch / Englisch		
SWS: 2		
Inhalte: Aktuelle Forschungsbeiträge aus den Bereich "Datenbanken und Informationssysteme".		
Literatur: Aktuelle Forschungsbeiträge		
Zugeordnete Lehrveranstaltungen:		

Seminar Informationssysteme für Master (Seminar)

Veranstaltung wird online/digital abgehalten.

Veranstaltungen wie SIGMOD, ICDE oder VLDB gehören zu den renommiertesten internationalen Konferenzen im Bereich Datenbanksysteme. Hier werden in Sessions und Workshops von internationalen Forscherteams aktuelle Ergebnisse zu einem breiten Themenspektrum von Streamverarbeitung, Big Data, Sozialen Netzwerken, Graphdatenbanken bis hin zu Concurrency und Hardwareunterstützung präsentiert. Im Zuge des Seminars sollen ausgewählte Beiträge dieser Konferenzen vorgestellt und in einem größeren Kontext eingebettet werden. Zudem sollen die Grundlagen wissenschaftlichen Arbeitens näher betrachtet werden.

Prüfung

Vortrag und schriftliche Ausarbeitung

Seminar

Modul INF-0251: Seminar Artificial Intelligence <i>Seminar Artificial Intelligence</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe17) Modulverantwortliche/r: PD Dr. Jonghwa Kim		
Inhalte: Das Seminar wird als Blockseminar Ende Juni für SS oder Mitte Dezember für WS stattfinden. Der Themenbereich für dieses Seminar wird jährlich unter Berücksichtigung neuer Trends in "Künstliche Intelligenz und Intelligente Systeme" neu festgelegt.		
Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet "Künstliche Intelligenz und Intelligente Systeme" selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen. Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an. Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.		
Schlüsselqualifikationen: Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten; Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewußtsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement		
Arbeitsaufwand: Gesamt: 120 Std. 30 Std. Seminar (Präsenzstudium) 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: in der Regel mind. 1x pro Studienjahr	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Artificial Intelligence Lehrformen: Seminar Sprache: Deutsch / Englisch SWS: 2 ECTS/LP: 4.0		

Inhalte:

Das Seminar wird als Blockseminar Ende Juni für SS oder Mitte Dezember für WS stattfinden. Der Themenbereich für dieses Seminar wird jährlich unter Berücksichtigung neuer Trends in "Künstliche Intelligenz und Intelligente Systeme" neu festgelegt.

Literatur:

aktuelle Forschungsliteratur

Prüfung

Seminar Artificial Intelligence

Schriftlich-Mündliche Prüfung

Modul INF-0274: Seminar Embedded Intelligence for Health Care and Wellbeing (Master) <i>Seminar Embedded Intelligence for Health Care and Wellbeing (Master)</i>		4 ECTS/LP
Version 1.0.0 (seit WS17/18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
Lernziele/Kompetenzen: After attending the seminar, students will be able to autonomously acquire and understand advanced problem statements, concepts, methods, approaches, techniques, and technologies in the field of Embedded Intelligence for Health Care and Wellbeing. They possess the scientific techniques, communication skills, and the ability to employ suitable media, to present understandingly a special topic in spoken and written, and to discuss and evaluate scientifically challenging themes from the field in a critical way. Furthermore, they can recognise logical structures of thinking and debating and employ them constructively. Participants can express themselves in a clear and understandable way and present scientific topics. They understand how to structure a talk, to focus it - also given a complex content - on the essential messages, and to communicate them in a suitable way. The lines of arguments and strategies in case of disturbances are applied by the students. Students know how to perform energetically, to cope with the presentation media and to use them interactively. They manage to orient a talk toward a certain audience, to motivate the listeners also over a longer duration, and to employ different methods of moderation. Key skills: Fundamentals of good scientific practice; Analytical-methodological competency; Time management; Literature research; Self-contained work with English technical literature; Communication skills; Ability to present (in written and spoken) practical and theoretical ideas in an understandable, confident, and convincing way; Writing a report in the markup language LaTeX; Evaluation of methods, technologies, and solutions w.r.t. different aspects; Quality awareness.		
Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 5.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Embedded Intelligence for Health Care and Wellbeing (Master) Lehrformen: Seminar Sprache: Deutsch / Englisch SWS: 2		
Inhalte: In the seminar Embedded Intelligence for Health Care and Wellbeing, recent research works in this field are going to be discussed. This comprises both the acquisition of data through sensors and (e.g., microphones or electrodes) and the analysis and the modelling of the data. One important aspect is also the practicability of modern deep learning methods. Health Care and Wellbeing applications reach from tracking of health states (e.g., epilepsy or depression) to personal assistance services. The participating students will work on a certain aspect, supervised by a research associate of the chair. They will summarise their results in a written report and an oral presentation. Topics: E-Health, M-Health, Sensor Signal Analysis, Vital Signs, Big Data.		

Literatur:

Wird vom Dozenten oder der Dozentin bekannt gegeben

Prüfung

Vortrag und schriftliche Ausarbeitung

Seminar

Modul INF-0314: Seminar IT-Infrastrukturen in der Medizin für Master <i>Seminar IT Infrastructure in Medical Information Systems for Master Students</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe19) Modulverantwortliche/r: Prof. Dr. Frank Kramer		
Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet IT-Infrastrukturen für die Translationale Medizinische Forschung selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen. Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an. Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.		
Schlüsselqualifikationen: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewusstsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten		
Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar IT-Infrastrukturen in der Medizin für Master Lehrformen: Seminar Sprache: Deutsch SWS: 2		
Inhalte: Aktuelle Themen der IT-Infrastrukturen in der Medizin		
Literatur: wird in der jeweiligen Kickoff-Veranstaltung vorgestellt.		
Zugeordnete Lehrveranstaltungen:		

Seminar IT-Infrastrukturen in der Medizin für Master (Seminar)

Veranstaltung wird online/digital abgehalten.

Es wird eine Kickoff-Veranstaltung mit Einzelheiten zum weiteren Ablauf des Seminars stattfinden.

Prüfung

Seminar IT-Infrastrukturen in der Medizin für Master

Schriftlich-Mündliche Prüfung

Modul INF-0320: Seminar Process Mining <i>Seminar Process Mining</i>		4 ECTS/LP
Version 1.0.0 (seit WS19/20) Modulverantwortliche/r: Prof. Dr. Robert Lorenz		
<p>Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, ein wissenschaftlich anspruchsvolles Thema aus dem Gebiet "Process Mining" selbstständig zu erarbeiten, dieses klar, verständlich und überzeugend in Schrift und Vortrag zu präsentieren und sachlich über Vorträge zu diskutieren.</p> <p>Sie verfügen über die dafür notwendige wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz neuer Medien.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Fertigkeit der Dokumentation und verständlichen, sicheren und überzeugenden Darstellung von Ideen, Konzepten und Ergebnissen; Kommunikationsfähigkeit; Fähigkeit zum Einsatz neuer Medien; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Qualitätsbewußtsein;Wissenschaftliche Methodik;</p>		
<p>Arbeitsaufwand: Gesamt: 120 Std. 45 Std. Vorbereitung von Präsentationen (Selbststudium) 45 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: Modul Process Mining (INF-0243) - Pflicht		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 3.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile
<p>Modulteil: Seminar Process Mining Lehrformen: Seminar Sprache: Deutsch SWS: 2 ECTS/LP: 4.0</p>
<p>Inhalte: Aktuelle Forschungsarbeiten im Bereich Process Mining: Process Discovery, Conformance Checking, Enhancement, Preprocessing of logs (clustering, filtering), Handling of Noise, Synthesis based methods, Process Mining and Data Mining, Statistical methods in Process Mining, case studies, tooling. Das Seminar eignet sich zur Vorbereitung auf Abschlussarbeiten und Projektmodule.</p>
<p>Literatur: Literatur hängt vom jeweiligen Thema ab und wird im Lauf des Seminars bereitgestellt.</p>

<p>Prüfung Seminar Process Mining Schriftlich-Mündliche Prüfung / Prüfungsdauer: 60 Minuten Bearbeitungsfrist: 2 Monate</p>

Modul INF-0331: Seminar Computational Intelligence (Master) <i>Seminar Computational Intelligence (Master)</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>After attending the seminar, students will be able to autonomously acquire and understand advanced problem statements, concepts, methods, approaches, techniques, and technologies in the field of Computational Intelligence. They possess the scientific techniques, communication skills, and the ability to employ suitable media, to present understandingly a special topic in spoken and written, and to discuss and evaluate scientifically challenging themes from the field in a critical way. Furthermore, they can recognise logical structures of thinking and debating and employ them constructively.</p> <p>Participants can express themselves in a clear and understandable way and present scientific topics. They understand how to structure a talk, to focus it - also given a complex content - on the essential messages, and to communicate them in a suitable way. The lines of arguments and strategies in case of disturbances are applied by the students. Students know how to perform energetically, to cope with the presentation media and to use them interactively. They manage to orient a talk toward a certain audience, to motivate the listeners also over a longer duration, and to employ different methods of moderation.</p> <p>Key qualifications: Fundamentals of good scientific practice; Analytical-methodological competency; Time management; Literature research; Self-contained work with English technical literature; Communication skills; Ability to present (in written and spoken) practical and theoretical ideas in an understandable, confident, and convincing way; Writing a report in the markup language LaTeX; Evaluation of methods, technologies, and solutions w.r.t. different aspects; Quality awareness.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 120 Std. 30 Std. Seminar (Präsenzstudium) 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Computational Intelligence (Master)		
Lehrformen: Seminar		
Sprache: Deutsch / Englisch		
SWS: 2		
Inhalte: Fuzzy Logic, Neural Networks, Evolutionary Computation, Learning Theory, Probabilistic Methods		
Literatur: To be announced by the lecturers.		
Zugeordnete Lehrveranstaltungen:		
Seminar Computational Intelligence (Bachelor & Master) (Seminar)		
<i>*Veranstaltung wird online/digital abgehalten.*</i>		
The specific topics are introduced and assigned to students in the kick-off meeting during the first week of the lecture period. During the first meetings, a tutorial on oral presentation techniques and literature research is given.		

Students are going to follow on their topics under the supervision of the scientific staff of the chair. By the end of the lecture period, a short oral presentation has to be given by each student. A written report (survey paper) must be handed in afterwards.

Prüfung

Seminar Computational Intelligence (Master)

Schriftlich-Mündliche Prüfung

Modul INF-0337: Seminar Embedded Systems (Master) <i>Seminar Embedded Systems (Master)</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen:</p> <p>Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet Embedded Systems selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen.</p> <p>Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an.</p> <p>Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikationen: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewußtsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 120 Std. 30 Std. Seminar (Präsenzstudium) 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p>		
<p>Voraussetzungen:</p> <p>keine</p>		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Embedded Systems (Master)		
Lehrformen: Seminar		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte:</p> <p>Im Seminar werden Themen aus dem Bereich Embedded Systems behandelt. Jeder Seminarteilnehmer erhält individuelle Literaturhinweise, die dann im Laufe des Seminars durch weitere eigenständig erarbeitete Referenzen ergänzt werden sollen. Abschluss des Seminars stellt eine schriftliche Ausarbeitung sowie ein Vortrag über das behandelte Thema dar.</p>		

Literatur:

individuell gegeben und Selbstrecherche

Zugeordnete Lehrveranstaltungen:

Seminar Embedded Systems (Master) (Seminar)

Veranstaltung wird online/digital abgehalten.

Alle wichtigen Informationen zum Seminar finden Sie im Seminar-Wiki unter es-augsburg.de/seminar/.

Prüfung

Seminar Embedded Systems (Master)

Schriftlich-Mündliche Prüfung

Modul INF-0342: Seminar Digital Health (Master) <i>Seminar Digital Health (Master)</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen:</p> <p>After attending the seminar, students will be able to autonomously acquire and understand advanced problem statements, concepts, methods, approaches, techniques, and technologies in the field of Digital Health. They possess the scientific techniques, communication skills, and the ability to employ suitable media, to present understandingly a special topic in spoken and written, and to discuss and evaluate scientifically challenging themes from the field in a critical way. Furthermore, they can recognise logical structures of thinking and debating and employ them constructively.</p> <p>Participants can express themselves in a clear and understandable way and present scientific topics. They understand how to structure a talk, to focus it - also given a complex content - on the essential messages, and to communicate them in a suitable way. The lines of arguments and strategies in case of disturbances are applied by the students. Students know how to perform energetically, to cope with the presentation media and to use them interactively. They manage to orient a talk toward a certain audience, to motivate the listeners also over a longer duration, and to employ different methods of moderation.</p> <p>Key skills: Fundamentals of good scientific practice; Analytical-methodological competency; Time management; Literature research; Self-contained work with English technical literature; Communication skills; Ability to present (in written and spoken) practical and theoretical ideas in an understandable, confident, and convincing way; Writing a report in the markup language LaTeX; Evaluation of methods, technologies, and solutions w.r.t. different aspects; Quality awareness.</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 120 Std.</p> <p>30 Std. Seminar (Präsenzstudium)</p> <p>90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 5.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Digital Health (Master)		
Lehrformen: Seminar		
Sprache: Deutsch / Englisch		
SWS: 2		
<p>Inhalte:</p> <p>In the seminar Digital Health, recent research works in this field are going to be discussed. This comprises both the acquisition of data through sensors and (e.g., microphones or electrodes) and the analysis and the modelling of the data. One important aspect is also the practicability of modern deep learning methods. Digital Health applications reach from tracking of health states (e.g., epilepsy or depression) to personal assistance services. The participating students will work on a certain aspect, supervised by a research associate of the chair. They will summarise their results in a written report and an oral presentation.</p> <p>Topics: E-Health, M-Health, Sensor Signal Analysis, Vital Signs, Big Data.</p>		

Literatur:

Wird vom Dozenten oder der Dozentin bekannt gegeben

Zugeordnete Lehrveranstaltungen:

Seminar Digital Health (Bachelor & Master) (Seminar)

Veranstaltung wird online/digital abgehalten.

The specific topics are introduced and assigned to students in the kick-off meeting during the first week of the lecture period. During the first meetings, a tutorial on oral presentation techniques and literature research is given. Students are going to follow on their topics under the supervision of the scientific staff of the chair. By the end of the lecture period, a short oral presentation has to be given by each student. A written report (survey paper) must be handed in afterwards.

Prüfung

Seminar Digital Health (Master)

Schriftlich-Mündliche Prüfung

Modul INF-0344: Seminar Software Engineering verteilter Systeme (MA) <i>Seminar Software Engineering of Distributed Systems (MA)</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet Software Engineering für verteilte Systeme selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen. Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an. Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikation: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewusstsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten</p>		
<p>Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: Frühere Veranstaltung "Seminar über Software Engineering verteilter Systeme (MA)" (INF-0039) darf nicht belegt worden sein wegen Überschneidungen.		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Software Engineering verteilter Systeme (MA)		
Lehrformen: Seminar		
Sprache: Deutsch		
SWS: 2		
Inhalte: Aktuelle Software Engineering-Themen aus Industrie und Forschung.		
Literatur: Wird in der jeweiligen Kickoff-Veranstaltung vorgestellt.		

Zugeordnete Lehrveranstaltungen:

Seminar Software Engineering verteilter Systeme (Master) (Seminar)

Veranstaltung wird online/digital abgehalten.

Bestandteil dieses Seminars sind fortgeschrittene Ansätze und Techniken im Bereich Software Engineering. Dies betrifft alle Phasen des Softwareentwicklungszyklus von der Anforderungsanalyse bis hin zum Testen. Modellierungstechniken sowie domänenspezifische Sprachen bilden einen Schwerpunkt des Seminars. Unter anderem werden in diesem Seminar Themen in Kooperation mit dem Kernkompetenzzentrum FIM vergeben.

Prüfung

Seminar Software Engineering verteilter Systeme (MA)

Schriftlich-Mündliche Prüfung

Modul INF-0346: Seminar Automotive Software and Systems Engineering (MA) <i>Seminar Automotive Software and Systems Engineering (MA)</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet Automotive Software & Systems Engineering selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen. Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an. Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikation: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewusstsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten</p>		
<p>Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)</p>		
<p>Voraussetzungen: Frühere Veranstaltung "Seminar Grundlagen des Software Engineering für Automotive Systems (MA)" (INF-0040) darf nicht belegt worden sein wegen Überschneidungen.</p>		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Automotive Software and Systems Engineering (MA)		
Lehrformen: Seminar		
Sprache: Deutsch		
SWS: 2		
Inhalte: Aktuelle Software Engineering-Themen aus Industrie und Forschung.		

Literatur:

Wird in der jeweiligen Kickoff-Veranstaltung vorgestellt.

Zugeordnete Lehrveranstaltungen:

Seminar Automotive Software and Systems Engineering (Master) (Seminar)

Veranstaltung wird online/digital abgehalten.

Bestandteil dieses Seminars sind Ansätze und Techniken im Bereich des Automotive Software Engineerings. Dies betrifft alle Phasen des System- und Softwareentwicklungszyklus von der Anforderungsanalyse bis hin zum Testen. Themen behandeln verschieden Aspekte daraus, u.a. Modellierungstechniken, domänenspezifische Sprachen, autonomes Fahren und Problemstellung durch den Einsatz von Multicore-Systemen.

Prüfung

Seminar Automotive Software and Systems Engineering (MA)

Schriftlich-Mündliche Prüfung

Modul INF-0348: Seminar Avionic Software and Systems Engineering (MA) <i>Seminar Avionic Software and Systems Engineering (MA)</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet Avionic Software & Systems Engineering selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen. Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an. Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.		
Schlüsselqualifikation: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewusstsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten		
Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)		
Voraussetzungen: Frühere Veranstaltung "Seminar Grundlagen des Software Engineering für Avionic Systems (MA)" (INF-0041) darf nicht belegt worden sein wegen Überschneidungen.		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Avionic Software and Systems Engineering (MA) Lehrformen: Seminar Sprache: Deutsch SWS: 2		
Inhalte: Aktuelle Software Engineering-Themen aus Industrie und Forschung.		

Literatur:

Wird in der jeweiligen Kickoff-Veranstaltung vorgestellt.

Prüfung

Seminar Avionic Software and Systems Engineering (MA)

Schriftlich-Mündliche Prüfung

Modul INF-0349: Seminar Menschzentrierte Künstliche Intelligenz <i>Seminar Human-Centered Artificial Intelligence</i>		4 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet "Mensch-zentrierte Künstliche Intelligenz" selbstständig zu erarbeiten und bezogen auf ein spezielles wissenschaftlich anspruchsvolles Thema aus dem genannten Gebiet zu bewerten.		
Schlüsselqualifikationen: Fertigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten;		
Arbeitsaufwand: Gesamt: 120 Std. 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium) 30 Std. Seminar (Präsenzstudium)		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile**Modulteil: Seminar Menschzentrierte Künstliche Intelligenz****Lehrformen:** Seminar**Sprache:** Deutsch**SWS:** 2**Inhalte:**

Themen aus dem Bereich "Mensch-zentrierte Künstliche Intelligenz"

Literatur:

Literaturhinweise werden bei der Vorbesprechung bekanntgegeben.

Zugeordnete Lehrveranstaltungen:**Seminar zu Menschzentrierte Künstliche Intelligenz (Seminar)****Veranstaltung wird online/digital abgehalten.**

Themen aus dem Bereich "Mensch-zentrierte Künstliche Intelligenz" Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet "Mensch-zentrierte Künstliche Intelligenz" selbstständig zu erarbeiten und bezogen auf ein spezielles wissenschaftlich anspruchsvolles Thema aus dem genannten Gebiet zu bewerten.

Schlüsselqualifikationen: Fertigkeit der sicheren und überzeugenden Darstellung von Ideen und Konzepten;

Nähere Informationen: <https://www.informatik.uni-augsburg.de/lehrstuehle/hcm/lectures/2020ss/mzki>**Prüfung****Seminar Menschzentrierte Künstliche Intelligenz**

Schriftlich-Mündliche Prüfung

Modul INF-0364: Seminar Software Engineering in sicherheitskritischen Systemen (MA) <i>Seminar Software Engineering in Safety- and Security-Critical Systems (MA)</i>		4 ECTS/LP
Version 1.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Nach dem Besuch des Seminars sind die Studierenden in der Lage, weiterführende Problemstellungen, Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet Software Engineering in sicherheitskritische Systemen und deren verwandten Disziplinen selbstständig zu erarbeiten, zu analysieren und bezogen auf das individuelle Seminarthema aus dem genannten Gebiet zu bewerten. Sie verfügen über die wissenschaftliche Methodik, Kommunikationsfähigkeit und Fähigkeit zum Einsatz entsprechender Medien, um ein spezielles Thema in Wort und Schrift klar und verständlich zu präsentieren und wissenschaftlich anspruchsvolle Themenstellungen aus dem genannten Gebiet kritisch und argumentativ zu diskutieren und zu bewerten. Außerdem können sie die logischen Strukturen des Denkens und Argumentierens erkennen und zielführend einsetzen. Die Teilnehmenden können klar und verständlich formulieren und Fachinhalte frei vortragen. Sie verstehen es, einen Vortrag klar und nachvollziehbar zu strukturieren und auch bei komplexen Inhalten den Vortrag auf wesentliche Botschaften auszurichten und diese verständlich zu vermitteln. Argumentationsketten und Lösungsstrategien bei Störungen wenden sie gekonnt an. Die Studierenden verstehen es, präsent aufzutreten und souverän mit gängigen Präsentationsmedien umzugehen und diese interaktiv einzusetzen. Sie schaffen es, einen Vortrag auf eine bestimmte Zielgruppe auszurichten und den Zuhörer auch bei längeren Vortragsdauern zu motivieren und verschiedene Moderationstechniken anzuwenden.</p> <p>Schlüsselqualifikation: Literaturrecherche; Eigenständiges Arbeiten mit englischsprachiger Fachliteratur; Analytisch-methodische Kompetenz; Wissenschaftliche Methodik; Grundsätze guter wissenschaftlicher Praxis; Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und zu deren Dokumentation; Fertigkeit zum logischen, abstrakten, analytischen und konzeptionellen Denken und formaler Argumentation; Qualitätsbewusstsein, Akribie; Kommunikationsfähigkeit; Zeitmanagement; Bewertung von Lösungsansätzen, Verfahren, Techniken und Technologien unter unterschiedlichen Gesichtspunkten</p>		
<p>Arbeitsaufwand: Gesamt: 120 Std. 30 Std. Seminar (Präsenzstudium) 90 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: unregelmäßig	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Seminar Software Engineering in sicherheitskritischen Systemen (MA)		
Lehrformen: Seminar		
Sprache: Deutsch		
SWS: 2		
Inhalte: Aktuelle Software Engineering-Themen aus Industrie und Forschung.		
Literatur: Wird in der jeweiligen Kickoff-Veranstaltung vorgestellt.		
Zugeordnete Lehrveranstaltungen:		

Seminar Software Engineering in sicherheitskritischen Systemen (Master) (Seminar)

Veranstaltung wird online/digital abgehalten.

Bestandteil dieses Seminars sind fortgeschrittene Ansätze und Techniken im Bereich Software Engineering. Dies betrifft alle Phasen des Softwareentwicklungszyklus von der Anforderungsanalyse bis hin zum Testen. Modellierungstechniken sowie domänenspezifische Sprachen bilden einen Schwerpunkt des Seminars. Unter anderem werden in diesem Seminar Themen in Kooperation mit dem Kernkompetenzzentrum FIM vergeben.

Prüfung

Seminar Software Engineering in sicherheitskritischen Systemen (MA)

Schriftlich-Mündliche Prüfung

Modul INF-0042: Projektmodul Softwaremethodiken für verteilte Systeme <i>Project Module Software Methodologies for Distributed Systems</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Bernhard Bauer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet Softwaremethodiken für verteilte Systeme und verfügen dort über tiefergehende Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik;</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 15 Std. Seminar (Präsenzstudium) 285 Std. Praktikum (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Projektmodul Softwaremethodiken für verteilte Systeme		
Lehrformen: Praktikum		
Sprache: Deutsch		
SWS: 1		
Inhalte: Aktuelle Forschungsthemen an der Professur Softwaremethodik für verteilte Systeme		
Literatur: Wird zu den jeweiligen Themen bereitgestellt.		
Zugeordnete Lehrveranstaltungen:		
Oberseminar zu Softwaremethodik für verteilte Systeme <i>*Veranstaltung wird online/digital abgehalten.*</i>		
Prüfung		
Projektabnahme, Vortrag, Abschlussbericht Praktikum		

Modul INF-0059: Projektmodul Theoretische Informatik <i>Project Module Theory of Computation</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Torben Hagerup		
Lernziele/Kompetenzen: Lernziele: Nach der Teilnahme am Projektmodul sind die Studierenden in der Lage, Problemstellungen höherer Komplexität auf dem Gebiet der theoretischen Informatik zu verstehen, und sie verfügen dort über tiefergehende Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren und Techniken der theoretischen Informatik in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung dabei auftretender fachlicher Probleme einzusetzen. Dadurch haben sie die nötigen Voraussetzungen, um eigene wissenschaftliche Leistungen zu erbringen und an die internationale Forschung anzuknüpfen. Darüber hinaus verfügen die Studierenden über die Kommunikationsfähigkeit, die Fähigkeit zur eigenständigen Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen der theoretischen Informatik zu diskutieren, Zwischenziele zu definieren, Lösungsvorschläge kritisch zu bewerten und eigene Ideen einzuordnen und zu präsentieren. Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Qualitätsbewusstsein, Akribie, Selbständige Arbeit, Zeitmanagement, eigenständige Literaturrecherche, Arbeit mit englischsprachiger Fachliteratur, Grundsätze guter wissenschaftlicher Praxis.		
Arbeitsaufwand: Gesamt: 300 Std. 285 Std. Praktikum (Selbststudium) 15 Std. Seminar (Präsenzstudium)		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Projektmodul Theoretische Informatik Lehrformen: Praktikum Sprache: Deutsch SWS: 1		
Inhalte: Autonome Mitarbeit an aktuellen Forschungsthemen.		
Zugeordnete Lehrveranstaltungen: Oberseminar Theoretische Informatik <i>*Veranstaltung wird online/digital abgehalten.*</i>		
Prüfung Vortrag und schriftliche Ausarbeitung Praktikum		

Modul INF-0072: Projektmodul Organic Computing <i>Project Module Organic Computing</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Jörg Hähner		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet "Organic Computing" und verfügen dort über tiefergehende Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik.</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 285 Std. Praktikum (Selbststudium) 15 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Projektmodul Organic Computing		
Lehrformen: Praktikum		
Sprache: Deutsch		
SWS: 1		
Inhalte: Mitarbeit an aktuellen Forschungsthemen.		
Literatur: In Abhängigkeit vom zu bearbeitenden Thema: <ul style="list-style-type: none"> • Paper • Buch • Handbuch 		
Zugeordnete Lehrveranstaltungen:		
Oberseminar Organic Computing <i>*Veranstaltung wird online/digital abgehalten.*</i> Weiterführende Themen aus dem Bereich Organic Computing		

Prüfung

Vortrag und Abschlussbericht

Praktikum

Modul INF-0080: Projektmodul Datenbanken und Informationssysteme <i>Project Module Databases and Information Systems</i>		10 ECTS/LP
Version 1.6.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Peter Michael Fischer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet der Datenbanken und Informationssysteme und verfügen dort über tiefergehende Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 15 Std. Seminar (Präsenzstudium) 285 Std. Praktikum (Selbststudium)</p>		
<p>Voraussetzungen: Modul Datenbanksysteme (INF-0073) - empfohlen Modul Suchmaschinen (INF-0077) - empfohlen Modul Analyzing Massive Data Sets (INF-0277) - empfohlen</p>		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Projektmodul Datenbanken und Informationssysteme Lehrformen: Praktikum Sprache: Deutsch / Englisch SWS: 1</p>		
Inhalte: Arbeiten an aktuellen Forschungs		
<p>Literatur:</p> <ul style="list-style-type: none"> • Aktuelle Forschungsbeiträge zum Thema Datenbanken und Big Data • Handbücher 		
<p>Zugeordnete Lehrveranstaltungen: Oberseminar Datenbanken und Informationssysteme <i>*Veranstaltung wird online/digital abgehalten.*</i></p>		

Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten an und kontaktieren direkt einen Mitarbeiter, dessen Forschung Sie interessiert: <https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/dbis/>
Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte!

Prüfung

Softwareabnahme, Vortrag, Abschlussbericht

Praktikum

Modul INF-0096: Projektmodul Multimedia Computing <i>Project Module Multimedia Computing</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Rainer Lienhart		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet des Multimedia Computings (z.B. Bild-, Video- und Tonverarbeitung bzw. Bild-, Video- und Tonsuche) und des maschinellen Sehens (Objekterkennung, Personendetektion, Posenschätzung von Menschen) und verfügen dort über tiefere Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik;</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 285 Std. Praktikum (Selbststudium) 15 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Projektmodul Multimedia Computing Lehrformen: Praktikum Sprache: Deutsch Angebotshäufigkeit: nach Bedarf SWS: 1</p>		
<p>Inhalte: Die konkrete Aufgabenstellung aus dem weitenläufigen Gebiet des Multimedia und maschinellen Sehens (Bild-, Video- und Tonverarbeitung, Objekterkennung, Suche in Bild-, Video- und Tonmaterial) wird jedes Jahr aktuell für jeden Studenten einzeln neu entworfen.</p>		
<p>Literatur: Literaturhinweise werden zum Beginn der Veranstaltung bekanntgegeben.</p>		
<p>Zugeordnete Lehrveranstaltungen: Oberseminar Multimedia Computing <i>*Veranstaltung wird online/digital abgehalten.*</i></p>		

Prüfung

**Vortrag mit Softwarerepräsentation; Ausarbeitung mit Softwaredokumentation; Erklärung des Quellcodes
(Code Review)**

Praktikum

Modul INF-0108: Projektmodul Lehrprofessur für Informatik <i>Project Module Teaching Professorship Informatics</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Robert Lorenz		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf einem der Gebiete "Concurrent Systems", "Petri Netze" oder "Process Mining" und verfügen dort über tiefere Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik;</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 15 Std. Seminar (Präsenzstudium) 285 Std. Praktikum (Selbststudium)</p>		
Voraussetzungen: Grundkenntnisse in einschlägigen Forschungsgebieten des Lehrstuhls		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Projektmodul Lehrprofessur für Informatik Lehrformen: Praktikum Sprache: Deutsch / Englisch SWS: 1</p>		
<p>Inhalte: Mitarbeit an der Entwicklung formaler Grundlagen und theoretischer Ergebnisse, dem Entwurf und der Programmierung unterstützender Softwaretools und der Evaluation von Ergebnissen und Konzepten in aktuellen Forschungsprojekten des Lehrstuhls aus den Bereichen "Concurrent Systems", "Petri Nets" oder "Process Mining". Details: https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/educu-inf/lehre/</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • J. Desel, W. Reisig, G. Rozenberg: Lectures on Concurrency and Petri Nets, Springer, Lecture Notes in Computer Science 3098, 2004 • Wil M. P. van der Aalst: Process Mining. Data Science in Action. Springer, 2016. 		
Zugeordnete Lehrveranstaltungen:		
Oberseminar zu Lehrprofessur für Informatik		

Veranstaltung wird online/digital abgehalten.

Das Oberseminar fasst Vorträge und Arbeitsbesprechungen zu Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unter Lehre (siehe <https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/educoinf/lehre/>) unsere Informationen zu Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen (unter Lehrveranstaltungen) an und kontaktieren direkt einen Mitarbeiter. Die alleinige Teilnahme am Oberseminar ist nicht möglich. Forschungsthemen des Lehrstuhls: Petri Nets, Process Mining, Concurrent Systems

Prüfung

Vortrag und Abschlussbericht

Praktikum

Modul INF-0137: Projektmodul Software- und Systems Engineering <i>Project Module Software- and Systems Engineering</i>		10 ECTS/LP
Version 1.1.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität aus dem Gebiet des Software und Systems Engineering und verfügen dort über tiefergehende Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Fertigkeit zum logischen, analytischen und konzeptionellen Denken • Eigenständige Arbeit mit auch englischsprachiger Fachliteratur • Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen • Qualitätsbewußtsein • Kommunikationsfähigkeit • Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen • Grundsätze guter wissenschaftlicher Praxis • Projektmanagementfähigkeiten • Wissenschaftliche Methodik 		
<p>Arbeitsaufwand: Gesamt: 300 Std. 285 Std. Praktikum (Selbststudium) 15 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Projektmodul Software- und Systems Engineering		
Lehrformen: Praktikum		
Sprache: Deutsch / Englisch		
SWS: 1		
Inhalte: Mitarbeit an aktuellen Forschungsthemen des Lehrstuhls		
Literatur: Abhängig von dem konkreten Projekt: Wissenschaftliche Papiere, Dokumentation		
Zugeordnete Lehrveranstaltungen: Oberseminar Software- und Systems Engineering		

Veranstaltung wird online/digital abgehalten.

Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen am Lehrstuhl für Softwaretechnik zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten an und kontaktieren Sie direkt einen Mitarbeiter, dessen Forschung Sie interessiert: <https://www.uni-augsburg.de/de/fakultaet/fai/isse/prof/swtse/teaching/students/> Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte!

Prüfung

Projektmodul Software- und Systems Engineering Projektabnahme

praktische Prüfung / Bearbeitungsfrist: 2 Monate

Modul INF-0165: Projektmodul Theorie verteilter Systeme <i>Project Module in Theory of Concurrent Systems</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Walter Vogler		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet "Theorie verteilter Systeme" und verfügen dort über tiefere Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 285 Std. Praktikum (Selbststudium) 15 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Projektmodul Theorie verteilter Systeme		
Sprache: Deutsch		
SWS: 1		
Inhalte: aktuelle Forschungsthemen in der Theorie verteilter Systeme		
Literatur: wissenschaftliche Papiere, evtl. Handbücher		
Zugeordnete Lehrveranstaltungen:		
Oberseminar zu Grundlagen Reaktiver Systeme <i>*Veranstaltung wird online/digital abgehalten.*</i>		
Prüfung		
Schriftliche Ausarbeitung Projektarbeit		

Modul INF-0170: Projektmodul Human-Centered Multimedia <i>Project Module Human-Centered Multimedia</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe13) Modulverantwortliche/r: Prof. Dr. Elisabeth André		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet "Human-Centered Multimedia" und verfügen dort über tiefere Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik;</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 15 Std. Seminar (Präsenzstudium) 285 Std. Praktikum (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Projektmodul Human-Centered Multimedia Lehrformen: Praktikum Sprache: Deutsch SWS: 1</p>		
Inhalte: Mitarbeit an aktuellen Forschungsthemen.		
Literatur: Literaturhinweise werden je nach Thema zu Beginn des Moduls gegeben.		
Zugeordnete Lehrveranstaltungen: Oberseminar Human-Centered Multimedia <i>*Veranstaltung wird online/digital abgehalten.*</i>		
Prüfung Projektmodul Human-Centered Multimedia praktische Prüfung		

Modul INF-0275: Projektmodul Embedded Intelligence for Health Care and Wellbeing <i>Project Module Embedded Intelligence for Health Care and Wellbeing</i>		10 ECTS/LP
Version 1.1.0 (seit WS17/18) Modulverantwortliche/r: Prof. Dr. Björn Schuller		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet der eingebetteten Systeme und der intelligenten Signalanalyse, insbesondere für Anwendungen in der Medizin- und Sportinformatik, und verfügen dort über tiefere Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik; Softwareentwicklung und -test.</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 285 Std. Praktikum (Selbststudium) 15 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Projektmodul Embedded Intelligence for Health Care and Wellbeing Lehrformen: Praktikum Sprache: Deutsch / Englisch SWS: 1</p>		
Inhalte: Autonome Mitarbeit an aktuelle Forschungsthemen.		
Literatur: Wissenschaftliche Veröffentlichungen; Handbücher; wird vom Lehrstuhl zur Verfügung gestellt.		
<p>Zugeordnete Lehrveranstaltungen: Oberseminar Embedded Intelligence for Health Care and Wellbeing <i>*Veranstaltung wird online/digital abgehalten.*</i></p>		

Prüfung

Vortrag und schriftliche Ausarbeitung

Praktikum

Modul INF-0328: Projektmodul IT-Infrastrukturen in der Medizin <i>Project Module IT Infrastructure in Medical Information Systems</i>		10 ECTS/LP
Version 1.0.0 (seit WS19/20) Modulverantwortliche/r: Prof. Dr. Frank Kramer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet der IT-Infrastrukturen in der translationalen medizinischen Forschung und verfügen dort über tiefere Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik.</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 285 Std. Praktikum (Selbststudium) 15 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		ECTS/LP-Bedingungen: Bestehen der Modulprüfung
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
<p>Modulteil: Projektmodul IT-Infrastrukturen in der Medizin Lehrformen: Praktikum Sprache: Deutsch SWS: 1</p>		
Inhalte: Aktuelle Forschungsthemen auf dem Gebiet der IT-Infrastrukturen in der translationalen medizinischen Forschung		
Literatur: wissenschaftliche Aufsätze, Handbücher		
<p>Zugeordnete Lehrveranstaltungen: Oberseminar IT-Infrastrukturen für die Translationale Medizinische Forschung <i>*Veranstaltung wird online/digital abgehalten.*</i> Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen und kontaktieren direkt einen Mitarbeiter, dessen Forschung Sie interessiert: https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/misit/ Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte!</p>		

Prüfung

Projektmodul IT-Infrastrukturen in der Medizin

praktische Prüfung

Modul INF-0340: Projektmodul Embedded Systems <i>Project Module Embedded Systems</i>		10 ECTS/LP
Version 1.0.0 (seit SoSe20) Modulverantwortliche/r: Prof. Dr. Sebastian Altmeyer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet der Embedded Systems und verfügen dort über tiefere Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 285 Std. Praktikum (Selbststudium) 15 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile
<p>Modulteil: Projektmodul Embedded Systems Lehrformen: Praktikum Sprache: Deutsch SWS: 1</p>
<p>Inhalte: Autonome Mitarbeit an aktuelle Forschungsthemen.</p>
<p>Literatur: wissenschaftliche Papiere, Handbücher</p>
<p>Zugeordnete Lehrveranstaltungen: Oberseminar Embedded Systems <i>*Veranstaltung wird online/digital abgehalten.*</i> Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten an und kontaktieren direkt einen Mitarbeiter, dessen Forschung Sie interessiert: www.es-augsburg.de Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte.</p>

Prüfung

Projektmodul Embedded Systems

praktische Prüfung

Modul INF-0374: Projektmodul Resource Aware Algorithmics <i>Project Module Resource Aware Algorithmics</i>		10 ECTS/LP
Version 1.0.0 (seit WS20/21) Modulverantwortliche/r: Prof. Dr. Tobias Mömke		
<p>Lernziele/Kompetenzen: Nach der Teilnahme am Projektmodul verstehen die Studierenden Problemstellungen höherer Komplexität auf dem Gebiet der Resource Aware Algorithmics und verfügen dort über tiefergehende Fachkenntnisse und Fähigkeiten. Sie können Konzepte, Methoden, Verfahren, Techniken und Technologien des genannten Gebiets in Forschungsprojekten entwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Darüber hinaus verfügen die Studierenden über die Team- und Kommunikationsfähigkeit, die Fähigkeit zur Literaturrecherche und die wissenschaftliche Methodik, um Problemstellungen auf dem Gebiet zu diskutieren, Zwischenziele zu definieren, sowie Zwischenergebnisse und innovative Ideen kritisch zu bewerten, einzuordnen, zu kombinieren und zu präsentieren.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständige Arbeit mit englischsprachiger Fachliteratur; Verständliche, sichere und überzeugende Präsentation von Ideen, Konzepten und Ergebnissen; Qualitätsbewußtsein; Kommunikationsfähigkeit; Fertigkeit der Zusammenarbeit in Teams und Verstehen von Teamprozessen; Grundsätze guter wissenschaftlicher Praxis; Projektmanagementfähigkeiten; Wissenschaftliche Methodik</p>		
<p>Arbeitsaufwand: Gesamt: 300 Std. 285 Std. Praktikum (Selbststudium) 15 Std. Seminar (Präsenzstudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile
<p>Modulteil: Projektmodul Resource Aware Algorithmics Lehrformen: Praktikum Sprache: Deutsch SWS: 1</p>
<p>Inhalte: Autonome Mitarbeit an aktuelle Forschungsthemen.</p>
<p>Literatur: wissenschaftliche Papiere, Handbücher</p>
<p>Zugeordnete Lehrveranstaltungen: Oberseminar Resource Aware Algorithmics <i>*Veranstaltung wird online/digital abgehalten.*</i> Das Oberseminar fasst Vorträge und Arbeitsbesprechungen zu Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unter Lehre (siehe https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/raa/studium-und-lehre/) unsere Informationen zu Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen (unter Lehrveranstaltungen) an und kontaktieren direkt einen Mitarbeiter. Die alleinige Teilnahme am Oberseminar ist</p>

nicht möglich. Forschungsthemen des Lehrstuhls: Approximationsalgorithmen, Online Algorithmen, Algorithmen für Big Data

Prüfung

Projektmodul Resource Aware Algorithmics

Portfolioprüfung

Modul INF-0003: Masterarbeit <i>Master's Thesis</i>		30 ECTS/LP
Version 1.0.0 (seit WS13/14) Modulverantwortliche/r: Alle Professorinnen und Professoren, die Veranstaltungen für diesen Studiengang anbieten.		
<p>Lernziele/Kompetenzen:</p> <p>Die Studierenden sind mit der wissenschaftlichen Methodik sowie Techniken der Literaturrecherche vertraut, sind in der Lage, unter Anleitung vertiefte praktische oder theoretische Methoden zur Bearbeitung eines vorgegebenen Themas einzusetzen und besitzen die Kompetenz, ein wissenschaftlich anspruchsvolles Problem der Informatik innerhalb einer vorgegebenen Frist weitgehend selbständig mit wissenschaftlichen Methoden zu bearbeiten sowie die Ergebnisse schriftlich und mündlich verständlich und überzeugend darzustellen.</p> <p>Außerdem verfügen die Studierenden über tiefgehende Fachkenntnisse und Fähigkeiten auf dem Gebiet der Masterarbeit, können Konzepte, Methoden, Verfahren, Techniken und Technologien dieses Gebiets in Forschungs- oder Anwendungsprojekten weiterentwickeln und sind fähig, innovative Methoden bei der Lösung von Problemen anzuwenden. Dadurch ist es ihnen möglich, an die internationale Forschung anzuknüpfen und ihren eigenen wissenschaftlichen Beitrag auf diesem Gebiet zu leisten. Die Studierenden haben die Fertigkeit zur Analyse und Strukturierung von Problemstellungen auf dem Gebiet der Masterarbeit und die Fertigkeit zur Entwicklung und Umsetzung von Lösungsstrategien, und kennen praxisrelevante Aufgabenstellungen. Sie kennen Vor- und Nachteile von Lösungsalternativen, können diese im jeweiligen Anwendungszusammenhang mit geeigneten Methoden wissenschaftlich aussagekräftig bewerten und geeignete Lösungsalternativen sicher auswählen. Die Studierenden haben die Fähigkeit, ihr Wissen auf dem Gebiet der Masterarbeit und verwandter Gebiete selbstständig zu erweitern und zu vertiefen.</p> <p>Darüber hinaus können die Studierenden weiterführende Strategien, Methoden und klare Vorstellungen entwickeln über ihre Ziele und Prioritäten, sie bewerten ihren persönlichen Arbeitsstil und schaffen eine effiziente Nutzung ihrer Ressourcen. Sie wenden Hilfsmittel und Techniken der Selbstorganisation an, die ihrem persönlichen Arbeitsstil entsprechen.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken, Team- und Kommunikationsfähigkeit, Durchhaltevermögen, schriftliche und mündliche Darstellung eigener (praktischer oder theoretischer) Ergebnisse, Fertigkeit der effizienten Ressourcennutzung, Einschätzung der Relevanz eigener Ergebnisse, Grundsätze guter wissenschaftlicher Praxis, Fähigkeit zur Selbstreflexion und eines reflektierten Arbeitsstils, Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von eigenen und fremden (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und deren Dokumentation, Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand:</p> <p>Gesamt: 900 Std. 900 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p>		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: 4.	Minimale Dauer des Moduls: 1 Semester
SWS: 0	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Masterarbeit		
Sprache: Deutsch		
Inhalte: entsprechend dem gewählten Thema		

Literatur:

Die Festlegung der Literatur erfolgt abhängig vom konkreten Thema der Arbeit in Absprache mit dem Betreuer.

Prüfung

Masterarbeit

Masterarbeit

Modul INF-0221: Anleitung zu wissenschaftlichen Arbeiten <i>Introduction to Scientific Work</i>		0 ECTS/LP
Version 1.0.0 (seit WS15/16) Modulverantwortliche/r:		
Lernziele/Kompetenzen: Die Teilnehmer wissen, wie sie an wissenschaftliche Arbeiten heran gehen, welche Vorgehensweise sie ans Ziel führt und welche Maßstäbe gelten, damit ihre Arbeit als wissenschaftlich angesehen wird.		
Bemerkung: Dies ist eine freiwillige Veranstaltung und gibt keine ECTS-Punkte!		
Arbeitsaufwand: Gesamt: 15 Std.		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester
SWS: 1	Wiederholbarkeit: keine	
Modulteile		
Modulteil: Anleitung zu wissenschaftlichen Arbeiten Sprache: Deutsch SWS: 1		
Inhalte: Begleitung bei der Anfertigung von Seminar-/Bachelor-/Master-/Diplomarbeiten und Dissertationen.		
Zugeordnete Lehrveranstaltungen: Anleitung zu wissenschaftlichen Arbeiten <i>*Veranstaltung wird online/digital abgehalten.*</i> Anleitung zu wissenschaftlichen Arbeiten <i>*Veranstaltung wird online/digital abgehalten.*</i> Anleitung zu wissenschaftlichen Arbeiten Software- und Systems Engineering <i>*Veranstaltung wird online/digital abgehalten.*</i>		

Modul INF-0222: Oberseminar Informatik <i>Graduate Seminar Computer Science</i>		0 ECTS/LP
Version 1.0.0 (seit WS15/16) Modulverantwortliche/r:		
Lernziele/Kompetenzen: Im Oberseminar werden wissenschaftliche Themen z.B. in Form von Abschlussarbeiten oder Vorträgen zu Praxis-/Forschungs-/Projektmodulen vorgestellt und diskutiert. Die Studierenden erhalten somit Einblicke in wissenschaftliches Arbeiten.		
Bemerkung: Dies ist eine freiwillige Veranstaltung und gibt keine ECTS-Punkte!		
Arbeitsaufwand: Gesamt: 30 Std. 30 Std. Seminar (Präsenzstudium)		
Voraussetzungen: keine		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester
SWS: 2	Wiederholbarkeit: keine	

Modulteile
Modulteil: Oberseminar Informatik Lehrformen: Seminar Sprache: Deutsch SWS: 2
Zugeordnete Lehrveranstaltungen: Oberseminar Datenbanken und Informationssysteme <i>*Veranstaltung wird online/digital abgehalten.*</i> Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten an und kontaktieren direkt einen Mitarbeiter, dessen Forschung Sie interessiert: https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/dbis/ Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte! Oberseminar Embedded Intelligence for Health Care and Wellbeing <i>*Veranstaltung wird online/digital abgehalten.*</i> Oberseminar Embedded Systems <i>*Veranstaltung wird online/digital abgehalten.*</i> Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten an und kontaktieren direkt einen Mitarbeiter, dessen Forschung Sie interessiert: www.es-augsburg.de Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte. Oberseminar Human-Centered Multimedia <i>*Veranstaltung wird online/digital abgehalten.*</i> Oberseminar IT-Infrastrukturen für die Translationale Medizinische Forschung <i>*Veranstaltung wird online/digital abgehalten.*</i>

Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen und kontaktieren direkt einen Mitarbeiter, dessen Forschung Sie interessiert: <https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/misit/> Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte!

Oberseminar Multimedia Computing

Veranstaltung wird online/digital abgehalten.

Oberseminar Organic Computing

Veranstaltung wird online/digital abgehalten.

Weiterführende Themen aus dem Bereich Organic Computing

Oberseminar Resource Aware Algorithmics

Veranstaltung wird online/digital abgehalten.

Das Oberseminar fasst Vorträge und Arbeitsbesprechungen zu Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unter Lehre (siehe <https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/raa/studium-und-lehre/>) unsere Informationen zu Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen (unter Lehrveranstaltungen) an und kontaktieren direkt einen Mitarbeiter. Die alleinige Teilnahme am Oberseminar ist nicht möglich. Forschungsthemen des Lehrstuhls: Approximationsalgorithmen, Online Algorithmen, Algorithmen für Big Data

Oberseminar Software- und Systems Engineering

Veranstaltung wird online/digital abgehalten.

Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen am Lehrstuhl für Softwaretechnik zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten an und kontaktieren Sie direkt einen Mitarbeiter, dessen Forschung Sie interessiert: <https://www.uni-augsburg.de/de/fakultaet/fai/isse/prof/swtse/teaching/students/> Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte!

Oberseminar Theoretische Informatik

Veranstaltung wird online/digital abgehalten.

Oberseminar zu Grundlagen Reaktiver Systeme

Veranstaltung wird online/digital abgehalten.

Oberseminar zu Lehrprofessur für Informatik

Veranstaltung wird online/digital abgehalten.

Das Oberseminar fasst Vorträge und Arbeitsbesprechungen zu Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unter Lehre (siehe <https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/educ-inf/lehre/>) unsere Informationen zu Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen (unter Lehrveranstaltungen) an und kontaktieren direkt einen Mitarbeiter. Die alleinige Teilnahme am Oberseminar ist nicht möglich. Forschungsthemen des Lehrstuhls: Petri Nets, Process Mining, Concurrent Systems

Oberseminar zu Softwaremethodik für verteilte Systeme

Veranstaltung wird online/digital abgehalten.

Oberseminar zur Mechatronik

Veranstaltung wird online/digital abgehalten.

Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn du ein solches Modul am Lehrstuhl für Mechatronik belegen willst, komm doch einfach auf uns zu (persönlich, Mail, zoom, ...). Wir freuen uns mit dir unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten zu diskutieren. Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte!

Oberseminar zur Produktionsinformatik

Veranstaltung wird online/digital abgehalten.

Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten an und kontaktieren direkt einen Mitarbeiter, dessen Forschung Sie interessiert: <https://www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/pi/lehre/fm-pm-seminar/> Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte!

Oberseminar zur Regelungstechnik

Veranstaltung wird online/digital abgehalten.

Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten an und kontaktieren direkt einen Mitarbeiter, dessen Forschung Sie interessiert: <https://www.informatik.uni-augsburg.de/lehrstuehle/rt/> Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte!

Oberseminar zur biomedizinischen Informatik

Veranstaltung wird online/digital abgehalten.

Das Oberseminar fasst Vorträge von Abschlussarbeiten und Praxis-/Forschungs-/Projektmodulen zusammen. Wenn Sie ein solches Modul belegen wollen, schauen Sie sich auf unserer Lehrstuhlhomepage unsere Forschungsthemen bzw. Themenvorschläge für studentische Arbeiten an und kontaktieren direkt einen Mitarbeiter, dessen Forschung Sie interessiert: www.uni-augsburg.de/de/fakultaet/fai/informatik/prof/bioinf/ Die alleinige Teilnahme am Oberseminar ist freiwillig und gibt keine ECTS-Punkte.