

Universität Paderborn



Informatik
**Veranstaltungs-
Kommentar**

Für

Informatik ▷ Bachelor
▷ Master

Computer Engineering

Ingenieurinformatik

Lehramt Informatik GyGe

Für das SoSe 16

Von der Fachschaft
Mathematik/Informatik



Inhaltsverzeichnis

1	Wichtige Informationen	3
1.1	Benutzerhinweise	3
1.2	Literaturangaben	3
1.3	Sprechstunden	3
1.4	Vollständigkeit	3
1.5	Internet	3
2	Mitarbeitende der Informatik	4
3	Weitere wichtige Adressen	7
4	Veranstaltungen	8
4.1	Übersicht	8
4.2	Informatik	12
5	Raum für Notizen	56
6	Ergebnisse der Veranstaltungskritik	57

Impressum

Herausgeber: Fachschaft Mathematik/Informatik
Universität Paderborn, Raum E1.311
Warburger Straße 100
33098 Paderborn
E-Mail: fsmi@uni-paderborn.de
Telefon: 05251 60-3260
Fax: 05251 60-3978

V.i.S.d.P.: Jan Beinke

ISSN: 1868-0690

Redaktion: Jan Beinke, Eduard Maas, Alex Wiens

Mitarbeitende: die Fachschaft (Korrekturlesen),
die Dozentinnen und Dozenten der Mathematik und der Informatik (Kommentare)

Auflage: 2⁶ Exemplare

1 Wichtige Informationen

1.1 Benutzerhinweise

zum Kopf:

Name der Veranstaltung

Dozent: Name des Dozenten

Büro: Raum

Sprechstunde: Zeit

1.2 Literaturangaben

Die Bücher in diesem Abschnitt sind Empfehlungen der Dozenten. Viele dieser Bücher sind in der Bibliothek zu finden, sodass ihr euch die Bücher dort erst ansehen und ausleihen könnt, bevor ihr viel Geld dafür ausgeben müsst. Auf Ebene 3 der Bibliothek befindet sich übrigens der Seminarapparat unserer Fachschaft. In diesem haben wir etwas Grundlagenliteratur zur Informatik und Mathematik gesammelt, welche wir für lesenswert halten.

1.3 Sprechstunden

Ein Großteil der Dozentinnen und Dozenten gibt keine feste Sprechstunde mehr an, sondern ist nach Vereinbarung zu sprechen, sowie vor und nach den Veranstaltungen. Daher findet Ihr nicht überall die Angabe einer Sprechstunde.

1.4 Vollständigkeit

Da nicht alle Lehrenden einen Veranstaltungskommentar abgegeben haben, ist das Verzeichnis der Veranstaltungen nicht vollständig!

1.5 Internet

Elektronische Informationen zum Vorlesungsangebot gibt es unter folgenden Adressen:

- <https://cs.uni-paderborn.de/studium/studienangebot/informatik/> – offizielle Webseite zum Studienangebot der Informatik
- <https://math.uni-paderborn.de/studium/studiengaenge/mathematik/> – offizielle Webseite zum Studium der Mathematik
- <http://webptool.cs.upb.de/> – aktuellster Stand der Vorlesungsplanung
- <https://paul.upb.de/> – offizielles Vorlesungsverzeichnis der Uni

Die Seiten der Fachschaft findet Ihr hier: <http://die-fachschaft.de/>

Jan Beinke, Eduard Maas, Alex Wiens
V-Kom-Redaktion für das SoSe 2016

2 Mitarbeitende der Informatik

Name	E-Mail	Telefon	Raum
Agne, Andreas	agne@upb.de	4348	O3.125
Ahlers, Ulrich	uli@hni.upb.de	6700	F2.320
Allwright, Michael	michael.allwright@upb.de	3320	ZM1.02-10
Antoniou, Paraskewi	antoniou@mail.upb.de	5394	O3.107
Auroux, Sebastian	sebastian.auroux@upb.de	5385	O3.167
Benter, Markus	benter@upb.de	6454	F1.122
Bewermeyer, Marion	m.bewermeyer@upb.de	6695	F2.411
Blobel, Johannes	blobel@ccs-labs.org	6494	FU.407
Blömer, Johannes	bloemer@upb.de	6651	F2.101
Blößl, Bastian	bloessl@ccs-labs.org	6494	FU.407
Blume, Bodo	bodo.blume@upb.de	1750	O3.173
Bobolz, Jan			
Bodden, Eric	eric.bodden@upb.de	3313	ZM1.02-09
Boschmann, Alexander	alexander.boschmann@upb.de	5397	O3.119
Böttcher, Stefan	stb@upb.de	6662	F2.217
Brandt, Sascha	myeti@mail.upb.de	6451	F1.203
Brauer, Sascha	sbrauer@upb.de	6653	F2.106
Brink, Christopher	christopher.brink@upb.de	3615	ZM1.02.07
Bujna, Kathrin	kabu@mail.upb.de	6627	F2.106
Bürger, Tanja	tanja.buerger@upb.de	5376	O3.155
Busa-Fekete, Robert,	busarobi@mail.upb.de	3348	O4.170
Buschmeyer, Carmen	carmen@upb.de	6412	F1.426
Cord-Landwehr, Andreas	cola@upb.de	6428	F1.119
Dietrich, Aljoscha			
Divband Soorati, Mohammad	divband@mail.upb.de	6468	F1.213
Domik, Gitta	domik@upb.de	6610	F2.204
Drächsler, Sevil	sevil.mehraghdam@upb.de	1755	O3.161
Drees, Maximilan	maxdrees@mail.upb.de	6433	F1.125
Dressler, Falko	dressler@ccs-labs.org	6510	F1.401
Eidens, Fabian	feidens@msopb.de	6650	F2.108
Eikel, geb. Hüllmann, Martina	martinah@upb.de	6705	F2.323
Engels, Gregor	engels@upb.de	3337	ZM1.03.45
Feldkord, Björn	bjoernf@mail.upb.de	6434	F1.125
Feldmann, Rainer.	obelix@upb.de	6720	F2.401
Feldotto, Matthias	feldi@mail.upb.de	6452	F1.203
Fischer, Matthias	mafi@upb.de	6466	F1.223
Gerking, Christopher	christopher.gerking@upb.de	3307	ZM1.02-14
Gmyr, Robert	gmyr@mail.upb.de	6704	F2.323
Gundelach, Sigrid	sigu@upb.de	6655	F2.207
Günther, Peter	peter.guenther@upb.de	6327	F2.111
Hagenauer, Florian	hagenauer@ccs-labs.org	6492	F1.319
Hamann, Heiko	Heiko.Hamann@upb.de	6465	F1.221
Hartel, Rita	rst@upb.de	6612	F2.209
Hauenschild, Wilfried	wilf@upb.de	5393	O2.146
Haupt, Jutta	jutta@upb.de	3312	ZM1.02.08

Name	E-Mail	Telefon	Raum
Heindorf, Stefan	heindorf@upb.de	5465-207	ZM1.03-07
Henzgen, Sascha	shenzgen@mail.upb.de	3345	O4.164
Heppner, Sabrina	sheppner@mail.upb.de	6323	F2.209
Ho, Nam	namh@mail.upb.de	5396	O3.122
Hüllermeier, Eyke	eyke@upb.de	1771	O4.258
Hülsmann, Adrian	klecks@upb.de	6623	F2.124
Jahn, Claudia	claudia.jahn@upb.de	6622	F2.104
Jakobs, Marie-Christine	mcjakobs@mai.upb.de	1767	O4.128
Juhnke, Jakob	jakob.juhnke@upb.de	6328	F2.108
Jung, Daniel	jungd@hni.upb.de	6433	F1.125
Karl, Holger	hkarl@mail.upb.de	5375	O3.158
Kastens, Uwe	uwe@upb.de	6686	F2.308
Kaufmann, Paul	paul.kaufmann@upb.de	1718	O3.116
Keil, Reinhard	Reinhard.Keil@upb.de	6411	F1.428
Kenter, Tobias	kenter@upb.de	4340	O3.119
Kleine Büning, Hans	kbcs1@upb.de	3360	O4.255
Klerx, Timo	timo.klerx@upb.de	3351	O4.146
Klingler, Florian	klingler@ccs-labs.org	6493	F1.319
Kolb, Christina	ckolb@mail.upb.de	6725	F2.406
Krämer, Julia-Desirée	juliadk@mail.upb.de	5388	O4.131
Kraus, Hans-Joachim	hajo.kraus@upb.de	6538	F1.410
Kreiss, Lydia	Lydia.Kreiss@upb.de	6501	F1.404
Krohn, Jörg-Peter	krohn@upb.de	3325	E1.103
Lange, Christina	chlange@mail.upb.de	1749	O4.213
Lefèvre, Jonas	jonas.levivre@upb.de	6732	F2.317
Lengeling, Elisabeth	elisabeth.lengeling@upb.de	3361	O4.155
Lettmann, Theodor	lettman@upb.de	3350	O4.173
Liske, Gennadij	utyf@mail.upb.de	6626	F2.111
Löken, Nils	nilo@upb.de	6632	F2.313
Lösch, Achim	achim.loesch@upb.de	5395	O3.122
Lücker, Steven Christopher	sluecker@mail.upb.de	6522	F1.107
Mäcker, Alexander	amaecker@mail.upb.de	6427	F1.119
Magenheim, Johann S.	jsm@upb.de	6341	F2.116
Malatyali, Manuel	malatya@upb.de	6462	F1.122
Maniera, Jürgen	sammy@upb.de	3326	ZM1.02.04
Margaritis, Melanie	melanie.margaritis@upb.de	6603	F2.201
Mehic, Ahmet	amehic@upb.de	3303	ZM1.02.12
Meisner, Sebastian	sebastian.meisner@upb.de	4347	O3.128
Melnikov, Vitaly	melnikov@mail.upb.de	3349	O4.158
Memedi, Agon			
Meyer auf der Heide, F.	fmadh@upb.de	6480	F1.301
Meyer, Christina	chmeyer@upb.de	6733	E1.106
			F2.416
Mohr, Felix	fmohr@mail.upb.de	3352	O4.149
Mohsen, Ahmadi Fahander	ahamadim@mail.upb.de	3353	O4.167
Monien, Burkhard	bm@upb.de	6707	F2.413
Müller, Kathrin geb. Bröker	kathyb@upb.de	6600	F2.416

Name	E-Mail	Telefon	Raum
Nabeel, Muhammad	nabeel@ccs-labs.org	6515	F1.412
Neugebauer, Jonas	jonas.neugebauer@upb.de	6611	F2.201
Oberhoff, Andreas	oberhoff@upb.de	6514	F1.419
Ohrndorf, Laura	laura.orndorf@mail.upb.de	6613	F2.201
Parruca, Donald	donald.parruca@upb.de	1754	O3.164
Paul, Adil	adilp@mail.upb.de	3346	O4.161
Peuster, Manuel	manuel.peuster@upb.de	4341	O3.149
Pfahler, Peter	peter@upb.de	6688	F2.311
Pfannschmidt, Karlson	kiudee@mail.upb.de	3346	O4.161
Platenius, Marie Christin	m.platenius@upb.de	5465/176	ZM1.02-13
Platzner, Marco	platzner@upb.de	5250	O3.207
Plessl, Christian	christian.plessl@upb.de	5399	O3.110
Rammig, Franz J.	franz@upb.de	6500	F1.414
Riebler, Heinrich	heinrich.riebler@upb.de	5382	O3.131
Riechers, Sören	soerenri@mail.upb.de	6427	F1.119
Roger, Irene	irene.roger@upb.de	6620	F2.122
Schäfer, Wilhelm	wilhelm@upb.de	3313	ZM1.02.09
Schäfermeyer, Petra	petral@upb.de	6481	F1.304
Scharfenbaum, Joachim	joscha@upb.de	3327	E1.101
Scheideler, Christian	scheideler@upb.de	6728	F2.326
Schild, Christian	christian.schild@upb.de	6416	F1.104
Schlatt, Elisabeth	schlatt@mail.upb.de	3764	O4.125
Schroeder, Ulf-Peter	ups@upb.de	6726	F2.409
Schubert, David	david.schubert@upbde	2306	ZM1.02-14
Schultz-Friese, Tobias	tsf@upb.de	6664	F2.224
Schwabe, Arne	arne.schwabe@upb.de	1756	O3.146
Selke, Harald	hase@upb.de	6413	F1.101
Setzer, Alexander	asetzer@mail.upb.de	6721	F2.403
Shaker , Ammar	ammar.shaker@upb.de	3345	O4.164
SHK-IRB/Anmeldung		3322	E1.303
SHK-IRB/Fürstenallee		6755	F2.521
SHK-IRB/Techniker		3318/4332	E1.303
Simon, Jens	simon@upb.de	1731	O2.164
Skopalik, Alexander	skopalik@mail.upb.de	6457	F1.209
Sommer, Christoph	sommer@ccs-labs.org	6495	F1.322
Stoll, Christa	stoll@upb.de	3339	O4.213
Strothmann, Thim	thim@mail.upb.de	6730	F2.317
Szwillus, Gerd	szwillus@upb.de	6624	F2.119
Teetz, Alexander	alexandwer.teetz@upb.de	5465-215	ZM1.03-10
Thissen, Thomas	tici@upb.de	6701	F2.320
Töws, Manuel	mtowes@upb.de	4272	O4.122
Travkin, Oleg	oleg82@upb.de	1709	O4.131
Utermöhle, Michael	mike@upb.de	6666	F2.224
Vaz, Gavin	gavin.vaz@upb.de	1717	O3.131
Wahby, Mostafa	mostafa.wahby@upb.de	6468	F1.213
Walther, Sven	sven.walther@upb.de	3763	O4.122
Wassing, Heinz-Georg,	koala@upb.de	6430	F1.122

Name	E-Mail	Telefon	Raum
Wawilow, Anastasia	apetkau@upb.de	6625	F2.124
Wegener, Friedhelm	fw@upb.de	3354	ZM1.03-04
Wehrheim, Heike	wehrheim@upb.de	4331	O4.225
Wiechers, Beatrix	wiechers@upb.de	3336	ZM1.03.46
Wiederhold, Cornelia	connyw@upb.de	6523	F1.101
Wiersema, Tobias	tobias.wiersema@upb.de	4343	O3.125
Wilke, Adrian	adrian.wilke@upb.de	6733	F2.416
Winkelnkemper, Felix	winfel@hni.upb.de	6410	F1.104
Wolters, geb. Bokermann, Dennis	dennis.bokermann@upb.de	5465-221	ZM1.03-07
Zahn, Gero	gero.zahn@upb.de	6519	F1.107

3 Weitere wichtige Adressen

Name	E-Mail	Telefon	Raum
Fachschaft Mathematik/Informatik	fsmi@upb.de	3260	E1.311
Mathe-Treff		3775	D3.331
Mathe-Lernzentrum		1856	J2.324
Prüfungssekretariat Mathematik:			
Stephanie Besler	besler@zv.upb.de	4230	C2.315
Prüfungssekretariat Informatik:			
Manuel Leßmann	lessmann@zv.uni-paderborn.de	5207	C2.222
Rechnerbetreuung Didaktik	intermax@upb.de	3758	D2.339
Rechnerbetrieb Mathematik	pem@math.upb.de	3494	D2.301
Rechnerbetreuung Informatik	IRB-Support@upb.de	3318	E1.303

4 Veranstaltungen

4.1 Übersicht

Vorlesungen, für die uns bis Redaktionsschluss keine Kommentare erreicht haben, sind in der folgenden Übersicht mit -- gekennzeichnet.

Vorlesungen der Informatik

Grundstudium Bachelorstudiengang

Kastens	Grundlagen der Programmiersprachen	(1.1)	12
Wehrheim	Grundlagen der Programmierung 2	(1.1)	13
Bodden	Softwaretechnikpraktikum	(1.2)	14
Böttcher	Grundlagen der Datenbanken	(1.3)	16
Scheideler	Datenstrukturen und Algorithmen	(2.2)	17
Platzner	Grundlagen der technischen Informatik	(3.1)	--
Karl	Konzepte und Methoden der Systemsoftware	(3.2)	18
Keil	Grundlagen der Mensch-Maschine-Wechselwirkung	(4.1)	19

Hauptstudium Bachelorstudiengang

Hüllermeier	Data Mining	(1.1,4.1)	20
Wehrheim	Softwaremodellierung mit formalen Methoden	(1.1)	22
Blömer	Komplexitätstheorie	(2.1)	23
Meyer auf der Heide	Methoden des Algorithmenentwurfs	(2.1)	25
Scheideler	Verteilte Algorithmen und Datenstrukturen	(2.1)	26
Platzner	Eingebettete Prozessoren	(3.1)	--
Szwillus	Gestaltung von Webauftritten	(4.1)	27

Proseminare

Fischer / Schroeder	Algorithmische Geometrie		28
Meyer	Modellbasierte Softwareentwicklung für software-intensive Systeme		29
Wehrheim	Moderne Programmiersprachen		--

Masterstudiengang

Bodden	Designing code analyses for large-scale software systems (DECA)	(1.1-1.6)	30
Hüllermeier	Machine Learning I	(1.4)	32
Kleine Büning	Propositional Proof Systems	(1.3,1.4,1.5)	33
Anjorin	Software Quality Assurance	(1.1,1.5,1.6)	--
Feldmann	Combinatorial Optimization	(2.1,2.2)	34
Fischer	Computational Geometry	(2.1,2.2)	36
Blömer	Cryptographic Protocols	(2.2,2.3)	37
Blömer	Cryptography: Provable Security	(2.2,2.3)	38
Meyer auf der Heide	Routing and Data Management in Networks	(2.1,2.2,2.4)	39
Karl	Future Internet	(3.1,3.3)	40
Plessl	Hardware/Software Codesign	(3.4,3.5,3.6)	--
Kleinjohann	Intelligenz in eingebetteten Systemen	(3.4,3.6)	41
Kaufmann	Metaheuristics for Hardware Evolution	(3.4,3.5)	42
Dressler	Network Simulation	(3.1,3.2,3.6)	43
Hamann	Swarm Robotics	(3.2,3.4,3.6)	44
Dressler	Vehicular Networking	(3.1,3.3,3.6)	45
Domik	Data and Information Visualization	(4.1)	46
Szwillus	Modelling User Interfaces	(4.5,4.7)	47

Seminare

Scheideler	Algorithmen für verteilte Systeme	(2.1,2.2)	48
Skopalik	Algorithmic Game Theory	(2.1,2.2)	49
Scheideler	PRISMA	(2.1,2.2)	--
Dressler	Advanced Wireless Networking	(3.1,3.3,3.6)	50
Hamann	Markow-Ketten und Anwendungen in der Schwarmintelligenz	(3.2,3.4,3.6)	--
Domik	Aktuelle Themen in der Visualisierung	(4.1,4.2,4.3,4.5,4.6)	--
Tauber	Assistive Technologies: Multi-modal Human-Computer Interaction	(4.1,4.2,4.3,4.5,4.6)	--

Projektgruppen

Projektgruppen die bereits im letzten Semester gestartet sind, werden nicht mit aufgeführt.

Blömer	Initialization Algorithms for Clustering		--
Oberthür, Engels	Projektgruppe: History in Paderborn App		51

Oberseminare

Böttcher	Datenbanken	(SWT&IS)	--
Engels	Informationssysteme	(SWT&IS)	--
Hüllermeier	Maschinelles Lernen	(SWT&IS)	--
Bodden	Softwaretechnik	(SWT&IS)	--
Wehrheim	Spezifikation und Modellierung	(SWT&IS)	--
Kleine Büning	Wissensbasierte Systeme	(SWT&IS)	--
Meyer auf der Heide	Algorithmen und Komplexität	(MuA)	52
Domik	Interactive 3D Graphics and Games	(MuA/MMWW)	--
Meyer auf der Heide	SFB 901 On The Fly Computing	(MuA)	53
Scheideler	Theoretische Informatik	(MuA)	--
Karl	Rechnernetze	(ESS)	--
Platzner	Technische Informatik	(ESS)	--
Dressler	Verteilte Eingebettete Systeme	(ESS)	--
Domik	Computergrafik, Visualisierung und Bildverarbeitung	(MMWW)	--
Keil	Informatik und Gesellschaft	(MMWW)	--
Szwilius	Mensch-Computer-Interaktion	(MMWW)	--

Lehrveranstaltungen für andere Studiengänge

Selke	Einführung in die Informatik für Geisteswissenschaftler	55
Bodden	Softwarepraktikum	14
Lettmann	Webbasierte Informationssysteme	54

Allgemeine Veranstaltungen

Plessl	PC ² Kolloquium: Kolloquium der Parallelverarbeitung	--
--------	---	----

4.2 Informatik

Grundlagen der Programmiersprachen

Dozent: Kastens

Büro: siehe HomePage

Sprechstunde: siehe HomePage

Inhaltsangabe

Zu Zielen und Inhalt siehe Beschreibung des Moduls I.1.1 Programmiertechnik:

<http://cs.uni-paderborn.de/studium/formalitaeten/modulhandbuch>

Literaturangaben

Siehe vollständiges Vorlesungsmaterial des Vorjahres:

<http://ag-kastens.upb.de/lehre/material/gps>

Verschiedenes

Hörerkreis:

siehe PAUL

Prüfungsform:

Klausur

weiterführende Veranstaltungen:

Programming Languages and Compilers
(Modul II.1.1 Softwaretechnik und
Informationssysteme)

Vorbesprechung:

erste Vorlesung

Modulzugehörigkeit:

Informatik Modul I.1.1

vorausgesetzte Kenntnisse:

Siehe Beschreibung des Moduls I.1.1 Programmiertechnik:

<http://cs.uni-paderborn.de/studium/formalitaeten/modulhandbuch>

nächster Wiederholungstermin:

in jedem Sommersemester

Homepage:

<http://ag-kastens.upb.de/lehre/material/gps/>

Grundlagen der Programmierung 2

Dozent: Wehrheim

Büro: O4.225

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Vorlesung baut auf den Kenntnissen aus "Grundlagen der Programmierung 1" auf und behandelt weiterführend die folgenden Themen:

Design Patterns

XML

Threads, Nebenläufigkeit und Synchronisation

Einführung in die graphische Programmierung mit Swing

Verschiedenes

Hörerkreis:

Bachelor Informatik, 2. Semester

Prüfungsform:

Klausur

vorausgesetzte Kenntnisse:

GP1

nächster Wiederholungstermin:

Sommersemester 2017

Softwaretechnikpraktikum

Dozent: Bodden

Büro: ZM1.02-09

Inhaltsangabe

Das Softwaretechnikpraktikum ist eine praxisorientierte Lehrveranstaltung inklusive Vorlesungen zum Thema Projektmanagement. Eine komplexe Softwareentwicklungsaufgabe wird im Team von ca. zehn Studierenden unter Verwendung von UML und Java bearbeitet. Schwerpunkt des Praktikums ist die Vermittlung von Erfahrungen mit der gruppenorientierten Softwareentwicklung unter Benutzung marktüblicher Werkzeuge, Methoden und Prozesse. Die Erstellung von Meilensteinplänen, ein teilweise durch die Studierenden zu übernehmendes Projektmanagement, sowie die Anfertigung von Kostenschätzungen und die Protokollierung des Aufwandes durch Stundenzettel sind wesentliche Bestandteile, um die Praxisnähe des Projekts sicherzustellen.

Literaturangaben

- **Helmut Balzert:** *Lehrbuch der Software-Technik: Basiskonzepte und Requirements Engineering*, Heidelberg: Spektrum, 2009
- **Helmut Balzert:** *Lehrbuch der Software-Technik: Entwurf, Implementierung, Installation und Betrieb*, Heidelberg: Spektrum, 2011
- **Helmut Balzert:** *Lehrbuch der Software-Technik: Softwaremanagement*,

Heidelberg: Spektrum, 2008

- **Daniel Galin:** *Software Quality Assurance: From theory to implementation*, Harlow: Pearson

- **Watts S. Humphrey:** *Three Dimensions of Process Improvement Part I: Process Maturity*, CrossTalk: The Journal of Defense Software Engineering, Februar 1998,

<http://static1.1.sqspcdn.com/static/f/702523/9167925/1288288753533/199802-Humphrey.pdf>

- **Watts S. Humphrey:** *Three Dimensions of Process Improvement Part II: The Personal Process*, CrossTalk: The Journal of Defense Software Engineering, März 1998,

<http://static1.1.sqspcdn.com/static/f/702523/9167977/1288288919673/199803-Humphrey.pdf>

- **Watts S. Humphrey:** *Three Dimensions of Process Improvement Part III: The Team Process*, CrossTalk: The Journal of Defense Software Engineering, April 1998,

<http://static1.1.sqspcdn.com/static/f/702523/9168059/1288289124207/199804-Humphrey.pdf>

- **Ian Sommerville:** *Software Engineering*, Harlow: Pearson

Verschiedenes

Hörerkreis:

Informatik Bachelor, Informatik
Bachelor mit Schwerpunkt Informatik

Modulzugehörigkeit:

I.1.2 Softwaretechnik

Prüfungsform:

Projektarbeit, Klausur

Leistungspunkte:

10

vorausgesetzte Kenntnisse:

Teilnahmevoraussetzung für das Softwaretechnikpraktikum ist ein erfolgreicher Prüfungsabschluss in den Veranstaltungen Grundlagen der Programmierung 1 u. 2 sowie Softwareentwurf.

weiterführende Veranstaltungen:

Modellbasierte Softwareentwicklung

nächster Wiederholungstermin:

Sommersemester 2017

Vorbesprechung:

erster Vorlesungstermin

Homepage:

<https://www.hni.uni-paderborn.de/swt/lehre/softwaretechnik-softwarepraktikum-ss-2016>

Grundlagen von Datenbanken

Dozent: Böttcher

Büro: F2.217

Sprechstunde: siehe Web

Inhaltsangabe

Relationales Datenmodell

Algebra

Kalkül

SQL

Anfrageoptimierung

Datenbankschema-Entwurf

Transaktionen

Literaturangaben

- **Kemper,Eickler:** *Datenbanksysteme,*

Verschiedenes

Hörerkreis:

ib-4 , winfo-4 , Lehramt

Prüfungsform:

Klausur

Modulzugehörigkeit:

Info , 1. Abschnitt

weiterführende Veranstaltungen:

XML Databases

Databases and Information Systems

nächster Wiederholungstermin:

SS2017

Datenstrukturen und Algorithmen

Dozent: Scheideler

Büro: F2.326

Sprechstunde: Do, 16-17 Uhr

Inhaltsangabe

Algorithmen bilden die Grundlage jeder Hardware und Software: ein Schaltkreis setzt einen Algorithmus in Hardware um, ein Programm macht einen Algorithmus für den Rechner verstehbar. Algorithmen spielen daher eine zentrale Rolle in der Informatik. Wesentliches Ziel des Algorithmenentwurfs ist die (Ressourcen-)Effizienz, d.h. die Entwicklung von Algorithmen, die ein gegebenes Problem möglichst schnell und mit möglichst geringem Speicherplatz lösen.

Untrennbar verbunden mit effizienten Algorithmen sind effiziente Datenstrukturen, also Methoden, große Datenmengen im Rechner so zu organisieren, dass Anfragen wie Suchen, Einfügen und Löschen, aber auch komplexere Anfragen effizient beantwortet werden können.

Die in dieser Veranstaltung vorgeschlagenen Entwurfs- und Analysemethoden für effiziente Algorithmen und Datenstrukturen sowie die grundlegenden Beispiele wie Sortierverfahren, dynamische Datenstrukturen und Graphenalgorithmien gehören zu den Grundlagen für die Algorithmenentwicklung und Programmierung in weiten Bereichen der Informatik.

Literaturangaben

- **Cormen, Leiserson, Rivest, Stein:** *Introduction to Algorithms*, MIT Press / McGraw-Hill, 3rd ed., ISBN: 0-262-53305-8
- **Cormen, Leiserson, Rivest:** *Algorithmen - Eine Einführung*, Oldenburg, ISBN: 3-486-27515-1
- **Ottmann, Widmeyer:** *Algorithmen und Datenstrukturen*, Spektrum Akademischer Verlag, ISBN: 3-321-29535-8
- **Kleinberg, Tardos:** *Algorithm Design*, Addison-Wesley, ISBN: 0-312-29535-8
- **Sedgewick:** *Algorithms in Java (parts 1-4)*, Addison-Wesley, ISBN: 0-201-36120-5

Verschiedenes

Hörerkreis:

i-b2, i-l2, ie2, im2, winf2

Prüfungsform:

Klausur

weiterführende Veranstaltungen:

- Grundlegende Algorithmen
- Methoden des Algorithmenentwurfs
- Verteilte Algorithmen und Datenstrukturen

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/fg-ti/lehre0/ss2016/dua.html>

Modulzugehörigkeit:

Info 1. Studienabschnitt, MUA

vorausgesetzte Kenntnisse:

keine

nächster Wiederholungstermin:

SS 2017

Konzepte und Methoden der Systemsoftware

Dozent: Karl

Büro: O3.158

Sprechstunde: Montag, 13 - 14 Uhr

Inhaltsangabe

Die Veranstaltung KMS betrachtet grundlegende Funktionen systemnaher Software, insbesondere die wesentlichen Funktionen, die ein Betriebssystem erbringen muss. Eine besondere Herausforderung ist dabei die Behandlung von Nebenläufigkeit, die in vielfältiger Form in modernen IT-Systemen auftritt. Grundlegende Techniken wie Isolation und Virtualisierung werden wir auf unterschiedlichen Abstraktionsebenen vorfinden und wiederholt anwenden.

Ziel der Vorlesung ist die Vermittlung solch allgemeiner Prinzipien, Konzepte, Methoden und Techniken, wie sie in komplexen Hardware/Software-Systemen mit Nebenläufigkeit vorzufinden sind. Die Studenten sollen die Gemeinsamkeiten erkennen können und die Prinzipien als grundlegend für das Fach verstehen. Sie sollen insbesondere in Entwurfssituationen diese Methoden sinnvoll einsetzen können.

Literaturangaben

Es gibt kein Buch, das alle in der Vorlesung behandelten Themen sinnvoll abdeckt. Empfehlenswert ist jedoch die aktuelle Ausgabe von:

- **Stallings, W.:** *Betriebssysteme: Prinzipien und Umsetzung*, Pearson Studium

Verschiedenes

Hörerkreis:

Bachelor

nächster Wiederholungstermin:

SS 2017

Modulzugehörigkeit:

Modul I.3.2

Homepage:

<http://www-old.cs.uni-paderborn.de/fachgebiete/fachgebiet-rechnernetze/lehre.html>

Grundlagen Mensch-Maschine Wechselwirkung

Dozent: Keil

Büro: F1.428

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Gebrauchstauglichkeit von Software ist heute eine Kernanforderung an die Produktgestaltung in der Informatik. Dabei geht es zum einen darum mit Hilfe geeigneter Gestaltungsmaßnahmen Verständnisprozesse bei Nutzern zu fördern und unnötige Belastungen bei der Arbeit mit Softwaresystemen zu vermeiden. Zum anderen ist ein methodisches Repertoire erforderlich, um schon während des Entwurfs die Gebrauchstauglichkeit sichern zu können. Dazu sind eine Reihe von fachlichen Grundlagen erforderlich, die von rechtlichen Anforderungen über physiologische und psychologische Grundlagen bis hin zu Methoden und Techniken der Systemgestaltung reichen.

Die Studierenden sollen in die Lage versetzt werden, grundlegende Problembereiche der Mensch-Rechner-Interaktion zu erkennen und sie konstruktiv gestaltend umzusetzen. Damit sollen sie zugleich anschlussfähiges Wissen erwerben, das vor allem für die Zusammenarbeit mit Designern und Ergonomen erforderlich ist, aber auch für den Diskurs mit Medienwissenschaftlern und Pädagogen hilfreich ist. Die erworbenen Kenntnisse und Fertigkeiten bilden zugleich die Grundlage für vertiefende Veranstaltungen im Bereich der Mensch-Maschine-Wechselwirkung wie z.B. Usability Engineering, Webdesign oder Assistierende Technologien, Barrierefreiheit.

Literaturangaben

Wird in der Veranstaltung bekanntgegeben.

Verschiedenes

Modulzugehörigkeit:

I.4.1 Grundlagen Mensch Maschine-
Wechselwirkung

vorausgesetzte Kenntnisse:

Grundsätzliche Kenntnisse über die Entwicklung von Softwaresystemen.

nächster Wiederholungstermin:

SS 17

Homepage:

koala.uni-paderborn.de

Data Mining

Dozent: Hüllermeier

Büro: O4.258

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Vorlesung gibt eine Einführung in Verfahren zur Wissensentdeckung mithilfe von Methoden zur systematischen Suche nach Mustern in Daten. Der Schwerpunkt liegt auf effizienten algorithmischen Ansätzen für potenziell sehr große Datenbestände.

1. Einleitung
2. Ähnlichkeit und Ähnlichkeitssuche
3. Häufigkeitsanalyse: Itemset Mining
4. Dimensionalitätsreduktion und Visualisierung
5. Clusteranalyse
6. Netzwerkanalyse
7. Link-Analyse
8. Data Mining auf Datenströmen
9. Verteilte Verarbeitung großer Datenmengen

Literaturangaben

- **A. Rajaraman, J. Leskovec, J.D. Ullman:** *Mining of Massive Datasets*, Cambridge University Press, 2012.
- **I.H. Witten, E. Frank:** *Data Mining: Practical Machine Learning – Tools and Techniques with Java Implementations*, Morgan Kaufmann, 2000.
- **J. Han, M. Kamber:** *Data Mining: Concepts and Techniques (2nd ed.)*, Morgan Kaufmann, 2006.
- **P.N. Tan, M. Steinbach, V. Kumar:** *Introduction to Data Mining*, Addison-Wesley, 2006.
- **D.J. Hand, H. Mannila, P. Smyth:** *Principles of Data Mining*, MIT Press, 2001.
- **T.A. Runkler:** *Data Mining: Methoden und Algorithmen intelligenter Datenanalyse*, Vieweg+Teubner, 2010.

Verschiedenes

Hörerkreis:

Bachelor Informatik, Mathematik, Wirtschaftsinformatik

Modulzugehörigkeit:

SWT II.1.1, MMWW II.4.1

Prüfungsform:

Klausur

Leistungspunkte:

4

vorausgesetzte Kenntnisse:

Grundlegende Kenntnisse in Mathematik (lineare Algebra, Statistik), Programmierung, Algorithmen und Datenstrukturen.

weiterführende Veranstaltungen:

Machine Learning I, Machine Learning II

nächster Wiederholungstermin:

Sommersemester 2017

Softwaremodellierung mit formalen Methoden

Dozent: Wehrheim

Büro: O4.225

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Formale Methoden sind Sprachen zur Modellierung/Spezifikation von Systemen. Ein Modell eines (Soft- oder Hardware) Systems beschreibt auf einer gewissen Abstraktionsebene die Funktionalität des Systems. Im Gegensatz zu (den meisten) Programmiersprachen besitzen formale Methoden eine genau festgelegte Semantik, d.h. eine mathematische Beschreibung der Bedeutung einer Spezifikation. Diese Festlegung der Semantik erlaubt es, das Systemmodell bereits vor der eigentlichen Implementierung formal zu analysieren und mögliche Fehler frühzeitig zu finden. In der Vorlesung sollen verschiedene formale Methoden eingeführt werden, die für unterschiedliche Systemarten geeignet sind. Für jede dieser formalen Methoden werden Semantik und Analysetechniken vorgestellt und Modellierungsbeispiele zur Illustration des Einsatzbereiches besprochen.

Am Anfang der Vorlesung wird es vorrangig um die Modellierung von Parallelität und Kommunikation gehen. Hier werden Petrinetze und die Prozessalgebra CCS vorgestellt. Danach werden wir uns mit zustandsbasierten Formalismen zur Spezifikationen von Daten und Operationen (Z und Object-Z) sowie Sprachen zur Beschreibung von zeitlichen Aspekten (Timed Automata) beschäftigen.

Literaturangaben

Skript

Verschiedenes

Hörerkreis:

Bachelor Informatik

Prüfungsform:

mündliche Prüfung

nächster Wiederholungstermin:

SS 2017

Modulzugehörigkeit:

Info 2. Studienabschnitt, SWT

vorausgesetzte Kenntnisse:

- Modellierung, insbesondere Aussagen- und Prädikatenlogik
- Automatentheorie

Homepage:

<http://wwwcs.uni-paderborn.de/cs/ag-wehrheim/lehre.html>

Komplexitätstheorie

Dozent: Blömer

Büro: F2.101

Sprechstunde: Mi, 11-12 Uhr

Inhaltsangabe

Die Komplexitätstheorie ist eine wichtige Ergänzung der Theorie der Algorithmen. Ihr Ziel ist es, zu verstehen, warum gewisse Berechnungsprobleme schwierig sind und diese anhand ihrer Schwierigkeit zu klassifizieren. Das bekannteste und wichtigste Beispiel ist die Theorie der NP-Vollständigkeit.

Inhaltliche Gliederung:

- Komplexitätsklassen, P vs. NP
- Reduktionen und Vollständigkeit
- Platzkomplexität
- Hierarchiesätze
- Relativierung und Orakel-Turingmaschinen
- Polynomialzeit-Hierarchie
- Probabilistische Komplexitätsklassen

Literaturangaben

Grundlagen und Vorkenntnisse

- **Michael R. Garey, David S. Johnson (2005):** *Computers and intractability*, New York: Freeman, ISBN 0-7167-1044-7, 0-7167-1045-5
- **Michael Sipser (2006):** *Introduction to the theory of computation*, Thomson Course Technology, ISBN 0-619-21764-2, 978-0-619-21764-8

Komplexitätstheorie

- **Sanjeev Arora, Boaz Barak (2009):** *Computational Complexity: A Modern Approach*, New York: Cambridge University Press, ISBN 0-521-42426-7
- **Oded Goldreich (2008):** *Computational complexity*, Cambridge Univ. Press, ISBN 978-0-521-88473-0
- **Christos H. Papadimitriou (1994):** *Computational complexity*, Addison Wesley, ISBN 0-02-015308-2, 0-201-53082-1
- **Ingo Wegener (2003):** *Komplexitätstheorie*, Springer, ISBN 3-540-00161-1
- **Marcus Schaefer and Christopher Umans (2002):** *Completeness in the Polynomial-Time Hierarchy: A Compendium*, SIGACT News

Verschiedenes

Hörerkreis:

Bachelor Informatik

Modulzugehörigkeit:

MuA II.2.1

Prüfungsform:

mündl. Prüfung

Leistungspunkte:

4

vorausgesetzte Kenntnisse:

Einführung in Berechenbarkeit, Komplexität
und formale Sprachen.

weiterführende Veranstaltungen:

Cryptography: Provable Security
Cryptographic Protocols
im Masterstudiengang

nächster Wiederholungstermin:

in einem Jahr

Methoden des Algorithmenentwurfs

Dozent: Meyer auf der Heide

Büro: F1.301

Sprechstunde: Mi, 13-14 Uhr

Inhaltsangabe

In dieser Vorlesung werden Techniken des effizienten Algorithmenentwurfs eingeführt. Im einzelnen werden die folgenden Themen behandelt:

- Greedy Algorithmen
- Divide & Conquer
- Dynamische Programmierung
- Approximations-Algorithmen
- Lokale Suche
- Randomisierte-Algorithmen
- Online-Algorithmen

Literaturangaben

- **Kleinberg, Tardos:** *Algorithm Design*, Addison-Wesley

Verschiedenes

Modulzugehörigkeit:

2. Studienabschnitt des BSc-Studiengangs,
Modul II 2.1

weiterführende Veranstaltungen:

Approximation Algorithms
Randomized Algorithms
Online Algorithms
Algorithmic Game Theory
Optimization I+II

Homepage:

<https://www.hni.upb.de/alg/teaching/>

Verteilte Algorithmen und Datenstrukturen

Dozent: Scheideler

Büro: F2.326

Sprechstunde: Do, 16-17 Uhr

Inhaltsangabe

Die Vorlesung wird eine Einführung in die Grundlagen der verteilten Algorithmen und Datenstrukturen geben. Folgende Themen werden dabei behandelt:

- Netzwerktheorie
- Designprinzipien für verteilte Algorithmen und Datenstrukturen
- Einführung in die verteilte Programmierung
- Prozessorientierte Datenstrukturen
(((zyklische) Listen, Clique und hypercubische Graphen)
- Informationsorientierte Datenstrukturen
(verteiltes Hashing, verteilte Stacks und Queues und verteilte Heaps)

Dabei wird ein besonderer Wert auf Datenstrukturen und Algorithmen gelegt, die selbst für sehr große verteilte Systeme noch effizient arbeiten und die selbststabilisierend sind.

Literaturangaben

Für die Vorlesung werden wir auf neueste Forschungsergebnisse zurückgreifen. Dadurch gibt es keine Bücher, die begleitend zur Vorlesung gelesen werden können, aber zur Vorlesung wird es einen ausführlichen Foliensatz geben.

Verschiedenes

Hörerkreis:

i-b6, ie, im, winf

Prüfungsform:

Mündliche Prüfung und Softwareprojekt

nützliche Parallelveranstaltungen:

-

nächster Wiederholungstermin:

SS 2017

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/fg-ti/lehre0/ss2016/vads.html>

Modulzugehörigkeit:

Info 2. Studienabschnitt, MuA

vorausgesetzte Kenntnisse:

Datenstrukturen und Algorithmen

weiterführende Veranstaltungen:

Fortgeschrittene verteilte Algorithmen und Datenstrukturen

Vorbesprechung:

-

Gestaltung von Webauftritten

Dozent: Szwillus

Büro: F2.119

Sprechstunde: siehe Webseite

Inhaltsangabe

Die Vorlesung gibt einen Überblick über die wichtigsten Probleme der Gestaltung von Webauftritten, die die Web Usability betreffen und behandelt zugehörige Methoden und Techniken zu ihrer Behandlung.

Gliederung:

- Web-Design-Probleme
- Gestaltung des Inhalts (Informationsarchitektur)
- Gestaltung der Seitenstrukturen
- Navigation
- Layout Grafik, Typografie
- Gestaltung für mobile Geräte

Falls Zeit bleibt, werden spezielle Themen behandelt wie

- Internationalisierung
- Personalisierung

Literaturangaben

Literatur wird in der Vorlesung bekannt gegeben.

Verschiedenes

Hörerkreis:

Informatiker BA 2. SA, Wirtschaftsinformatiker, Medienwissenschaftler, Lehramtsstudenten, Ingenieurinformatiker

Modulzugehörigkeit:

Informatik 2. SA Mensch-Maschine-Wechselwirkung (MMWW)

Prüfungsform:

Klausur, Teilnahme an den Übungen

Leistungspunkte:

4

vorausgesetzte Kenntnisse:

Grundstudium des Informatik-Bachelorstudiums, aber insbesondere die Veranstaltung „Grundlagen der Mensch-Maschine-Wechselwirkung“. Dies ist aber keine formale Voraussetzung

nächster Wiederholungstermin:

SS 16

Vorbesprechung:

keine

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/fg-mci/lehre/ss-2016/gestaltung-von-webauftritten.html>

Proseminar: Algorithmische Geometrie

Dozent: Matthias Fischer, Ulf-Peter Schroeder

Büro: F1.223

Sprechstunde: nach Vereinbarung

Inhaltsangabe

In Proseminar werden Vorträge zu Themen aus der Algorithmischen Geometrie angeboten. Die Vorträge entsprechen Kapiteln aus dem Buch "Computational Geometry: Algorithms and Applications" Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars.

Es werden unter anderem behandelt:

- Schnitt von Liniensegmenten
- Polygontriangulierung
- Lineares Programmieren
- Bereichsuche
- Voronoi Diagramme und Delaunay Triangulierungen
- Konvexe Hüllen
- BSP Bäume
- Sichtbarkeitsgraphen
- Quadtrees
- Bewegungsplanung

Literaturangaben

- **Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars:** *Computational Geometry: Algorithms and Applications*,

Verschiedenes

Hörerkreis:

Bachelor Informatik

Prüfungsform:

gelungener Seminarvortrag und entsprechende Ausarbeitung

Homepage:

<http://www.hni.upb.de/en/alg/teaching>

Modulzugehörigkeit:

2. Studienabschnitt des BSc-Studiengangs

Vorbesprechung:

am ersten Veranstaltungstag, weitere Informationen auf der Webseite

Proseminar: Modellbasierte Softwareentwicklung für software-intensive Systeme

Dozent: Meyer

Büro: ZM1.02-36

Sprechstunde: auf Anfrage

Inhaltsangabe

Intelligente Technische Systeme (ITS) beruhen auf einer Symbiose von Informatik und Ingenieurwissenschaften und passen sich an ihre Umgebung und die Wünsche ihrer Anwender an. Zudem können sie mit unerwarteten Situationen umgehen und agieren vorausschauend auf der Basis von Erfahrungswissen. Solche Systeme werden z. B. im Haushalt, in der Produktion, im Handel oder auf der Straße eingesetzt.

ITS sind softwareintensive Systeme, denn ihre innovativen Fähigkeiten werden maßgeblich mithilfe von Software realisiert. Diese übernimmt komplexe Steuerungs-, Koordinations- und Berechnungsaufgaben, welche lokal oder auf Cloud-Computing-Plattformen ausgeführt werden.

Im Rahmen des Seminars wird der Stand der Forschung bezüglich der systematischen Entwicklung und Qualitätssicherung von Software für softwareintensive Systeme aufgearbeitet. Dabei werden insbesondere modellbasierte oder modellgetriebene Ansätze für die verschiedenen Entwicklungsphasen wie z. B. Anforderungserhebung, Analyse und Entwurf, Implementierung sowie Verifikation und Validierung behandelt.

Verschiedenes

Hörerkreis:

Bachelor Informatik/Ingenieurinformatik

Modulzugehörigkeit:

Bachelor Informatik 2. Studienabschnitt Modul II.5.1

Prüfungsform:

Schriftliche Ausarbeitung, Vortrag, mündl. Prüfung

Leistungspunkte:

3

vorausgesetzte Kenntnisse:

Softwaretechnikpraktikum

Vorbesprechung:

erste Semesterwoche; Termin wird noch bekannt gegeben

Vorlesungen: Softwareentwurf, Grundlagen der Programmierung

Empfohlen: Vorlesung Modellbasierte Softwareentwicklung

Designing code analyses for large-scale software systems (DECA)

Dozent: Bodden

Büro: ZM1.02-09

Inhaltsangabe

Statische Codeanalysen dienen dazu, automatisiert Fehler und Schwachstellen im Programmcode aufzufinden. Zu diesem Zwecke suchen sie nach bekannten Fehlermustern. In dieser Vorlesung wird erklärt, wie man solche Codeanalysen entwirft, die inter-prozedural sind, also das komplette Programm betrachten, über die Grenzen einzelner Prozeduren hinweg. Der Entwurf solcher Analysen gestaltet sich deshalb sehr schwierig, weil die Analysen oft Millionen von Programmstatements gleichermaßen präzise aber auch effizient verarbeiten müssen. Es werden außerdem Beispielsanalysen aus dem Bereich der IT-Sicherheit besprochen.

Behandelte Themen:

1. Intra-prozedurale Datenflussanalyse
2. Algorithmen zur Konstruktion von Callgraphen
3. Kontext-insensitive inter-procedurale Datenflussanalyse
4. Kontext-Sensitivität mittels des Call-Strings-Ansatzes
5. Value-basierte Kontexte
6. Kontext-Sensitivität mittels des Funktionalen Ansatzes
7. Distributive Probleme effizient mit den IFDS- und IDE-Frameworks lösen
8. Aktuelle Herausforderungen in der inter-prozeduralen Datenflussanalyse

Während der gesamten Veranstaltung werden Anwendungsbeispiele aus dem Gebiet der Softwaresicherheit diskutiert.

Literaturangaben

- **Thomas Reps, Susan Horwitz, and Mooly Sagiv:** *Precise interprocedural dataflow analysis via graph reachability*, POPL '95
- **Shmuel Sagiv, Thomas W. Reps, and Susan Horwitz:** *Precise Interprocedural Dataflow Analysis with Applications to Constant Propagation*, TAPSOFT '95
- **Akash Lal, Thomas Reps, and Gogul Balakrishnan:** *Extended weighted pushdown systems*, CAV 2005
- **Nomair A. Naeem, Ondrej Lhoták, and Jonathan Rodriguez:** *Practical extensions to the IFDS algorithm*, CC 2010
- **Yannis Smaragdakis, Martin Bravenboer, and Ondrej Lhoták:** *Pick your contexts well: understanding object-sensitivity*, POPL 2011
- **Eric Bodden:** *Inter-procedural data-flow analysis with IFDS/IDE and Soot*, SOAP 2012

- **Rohan Padhye, Uday P. Khedker:** *Interprocedural Data Flow Analysis in Soot using Value Contexts*, SOAP 2013

Verschiedenes

Hörerkreis:

Master Informatik

Prüfungsform:

mündliche Prüfung

vorausgesetzte Kenntnisse:

Der vorherige Besuch der Veranstaltung Software Analysis wird empfohlen ist aber nicht zwingend erforderlich. Ein gutes Verständnis von Java und den Prinzipien objektorientierter Programmierung ist hilfreich.

nächster Wiederholungstermin:

Sommersemester 2017

Modulzugehörigkeit:

III.1.1 Modellbasierte Softwareentwicklung,
III.1.5 Analytische Methoden des Software Engineering

Leistungspunkte:

4

weiterführende Veranstaltungen:

in Planung

Homepage:

<https://www.hni.uni-paderborn.de/swt/lehre/deca>

Machine Learning I (in English)

Dozent: Hüllermeier

Büro: O4.258

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Due to the ever increasing amount of data that is routinely produced in our information society, the topic of machine learning has become increasingly important in the recent years, not only as a scientific discipline but also as a key technology of modern software and intelligent systems. This lecture provides an introduction to the topic of machine learning, with a specific focus on supervised learning for classification and regression. The lecture covers theoretical foundations of generalisation as well as practical topics and concrete learning algorithms.

1. Introduction
2. The Learning Problem
3. Training versus Testing
4. The Linear Model
5. Non-Linear Methods
6. Overfitting

Literaturangaben

- **Y.S. Abu-Mostafa, M. Magdon-Ismail, H.T. Lin.:** *Learning from Data*, AMLBook, 2012.
- **P. Flach.:** *Machine Learning*, Cambridge Univ. Press, 2012.
- **E. Alpaydin.:** *Machine Learning*, Oldenbourg, 2008.
- **C.M. Bishop.:** *Pattern Recognition and Machine Learning*, Springer, 2006.

Verschiedenes

Hörerkreis:

Master Informatik, Mathematik, Wirtschaftsinformatik

Modulzugehörigkeit:

SWT III.1.4

Prüfungsform:

Mündliche Prüfung

Leistungspunkte:

4

vorausgesetzte Kenntnisse:

Basic knowledge in mathematics (linear algebra, statistics), programming and algorithms.

weiterführende Veranstaltungen:

Machine Learning II

nächster Wiederholungstermin:

Sommersemester 2017

Propositional Proof Systems

Dozent: Kleine Büning

Büro: O4.255

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Vorlesung behandelt ausgewählte Themen aus dem Bereich der Aussagenlogik und ihre algorithmische Umsetzung. Ausgehend vom klassischen NP-vollständigen Problem SAT werden effiziente Erfüllbarkeitsalgorithmen, effizient entscheidbare Teilklassen der Aussagenlogik, verschiedene Beweiskalküle und ihre Komplexität angesprochen. Als Erweiterung der Aussagenlogik wird die Klasse der quantifizierten Booleschen Formeln vorgestellt und die Q-Resolution als kanonische Erweiterung der Resolution zur ihrer maschinellen Entscheidung.

Die Vorlesung wird in Englisch gehalten.

Literaturangaben

- **Kleine Büning, Lettmann:** *Propositional Logic: Deduction and Algorithms*, Cambridge University Press 1999

Verschiedenes

Hörerkreis:

Informatik Master, LSII, Winfo

Prüfungsform:

mündliche Prüfung

vorausgesetzte Kenntnisse:

Bachelor; nützlich: Kenntnisse aus der Vorlesung „Grundlagen wissensbasierter Systeme“

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/fg-kleine-buening/lehre/propositional-proof-systems.html>

Modulzugehörigkeit:

Modul III.1.4 (SWT&IS Wissensbasierte Systeme)

Leistungspunkte:

4 ECTS

nächster Wiederholungstermin:

offen

Combinatorial Optimization (in English)

Dozent: Feldmann

Büro: F2.401

Sprechstunde: nach Vereinbarung

Inhaltsangabe

In this course, we start with a brief repetition of the linear programming problem (LPs) and of algorithms to solve LPs. Then we'll give an overview of the theory and practice of integer programming (IP). We will describe the primary computational components of a modern integer program solver: methods for bounding, branching, and search.

On our way we will use Cplex as a standard solver for linear or integer programs.

Keywords:

- Linear programs (LP)
- Simplex algorithm
- Mixed integer linear programs (MIP)
- Algorithms for NP-hard optimization problems
- Matroids
- Totally unimodular Matrices
- Polyhedral Approaches

Literaturangaben

- **Alexander Schrijver:** *Combinatorial Optimization: Polyhedra and Efficiency*, Springer, 2003
- **Vasek Chvatal:** *Linear Programming*, Freeman 1983
- **Alexander Schrijver:** *Theory of Linear and Integer Optimization*, Wiley, 1999
- **Vangelis Th. Paschos (Ed.):** *Concepts of Combinatorial Optimization*, Wiley, 2010
- **Dimitris Bertsimas, Robert Weismantel.:** *Optimization over Integers*, Belmont, MA: Dynamic Ideas, 2005.

Verschiedenes

Hörerkreis:

Computer Science Master

Modulzugehörigkeit:

Models and Algorithms, Modules III.2.1, III.2.2.

Prüfungsform:

Oral exam (as part of a module exam)

Leistungspunkte:

4

vorausgesetzte Kenntnisse:

Basic knowledge from linear algebra, graph theory and algorithms is expected.

Knowledge from the lecture "Optimization" of the computer science bachelor program is helpful.

Vorbesprechung:

none

nächster Wiederholungstermin:

to be determined

Homepage:

in PAUL

Computational Geometry

Dozent: Fischer

Büro: F1.223

Sprechstunde: nach Vereinbarung

Inhaltsangabe

The lecture provides topics of Computational Geometry. Topics include: Voronoi diagrams, Epsilon networks and VC-dimension, algorithmic motion planning for robots, visibility in polygons, convex hull, lower contour of line segments and functions, Sweep method and applications, geometric data structures: dynamization, kd-trees, range trees, priority search tree.

Literaturangaben

- **Rolf Klein:** *Algorithmische Geometrie*,
- **Jiri Matousek:** *Lectures on Discrete Geometry*,
- **Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars:** *Computational Geometry: Algorithms and Applications*,

Verschiedenes

Hörerkreis:
Informatik Master

Modulzugehörigkeit:
III.2.1, III.2.2

vorausgesetzte Kenntnisse:
Advanced understanding of data structures and algorithms. For example lectures fundamental algorithms, data structures and algorithms.

Homepage:
<https://www.hni.upb.de/alg/teaching>

Cryptographic Protocols

Dozent: Blömer

Büro: F2.101

Sprechstunde: Mi, 11-12 Uhr

Inhaltsangabe

We will discuss authentication schemes, protocols for identification and their variants. Afterwards, we will go into zero knowledge protocols. Finally, we will build numerous cryptographic primitives upon these techniques.

In the first half of the semester there is a course on the foundations of provably secure cryptography.

Literaturangaben

- **O. Goldreich:** *Foundations of Cryptography I*, Cambridge University Press
- **Katz, Lindell:** *Introduction to Modern Cryptography*, Chapman & Hall/CRC

Verschiedenes

Hörerkreis:

Masterstudiengang Informatik

Prüfungsform:

mündliche Prüfung

vorausgesetzte Kenntnisse:

Einführung in die Kryptographie

nächster Wiederholungstermin:

SS2017

Modulzugehörigkeit:

Modul III.2.3 und III.2.2

Leistungspunkte:

4

nützliche Parallelveranstaltungen:

Cryptography: Provable Security

Cryptography: Provable Security

Dozent: Blömer

Büro: F2.101

Sprechstunde: Mi, 11-12 Uhr

Inhaltsangabe

We discuss several advanced security concepts like semantic security and plaintext indistinguishability. We also describe several techniques to design cryptosystems that meet these strong security requirements. In particular, we discuss one-way functions, pseudorandom generators, and pseudorandom functions. Finally, we compare the security of cryptosystems used in practice (like AES and RSA) with advanced security concepts.

In the second half of the semester there will be a course on cryptographic protocols.

Literaturangaben

- **O. Goldreich:** *Foundations of Cryptography I*, Cambridge University Press
- **Katz, Lindell:** *Introduction to Modern Cryptography*, Chapman & Hall/CRC

Verschiedenes

Hörerkreis:

Masterstudiengang Informatik

Prüfungsform:

mündliche Prüfung

vorausgesetzte Kenntnisse:

Einführung in die Kryptographie

nächster Wiederholungstermin:

SS 2017

Modulzugehörigkeit:

Modul III.2.3 und III.2.2

Leistungspunkte:

4

nützliche Parallelveranstaltungen:

Cryptographic Protocols

Routing and Data Management in Networks

Dozent: Meyer auf der Heide

Büro: F1.301

Sprechstunde: Mi, 13-14 Uhr

Inhaltsangabe

Routing and data management are basic tasks to be solved in order to allow efficient use of large networks like, e.g., the Internet, peer-to-peer systems, or wireless mobile ad-hoc networks. This lecture deals with algorithms and their analysis for routing and data management in such systems, and especially describes methods for handling their dynamics (movement of nodes, joins and leaves of nodes).

Literaturangaben

Werden auf der Webseite angegeben.

Verschiedenes

Hörerkreis:

Master Informatik

Modulzugehörigkeit:

MuA 3.SA Modul III.2.1, III.2.2, III.2.4

Prüfungsform:

voraussichtlich mündliche Prüfung

vorausgesetzte Kenntnisse:

Bachelorstudium

Homepage:

<http://www.hni.upb.de/en/alg/teaching>

Future Internet

Dozent: Karl

Büro: O3.158

Sprechstunde: Montag, 13 - 14 Uhr

Inhaltsangabe

This lecture is an attempt to capture current developments in Internet research. It does not follow a standard agenda, it is heavily focused on research and goes over various, not necessarily well connected topics. For parts of the lecture, we will follow a textbook on network virtualization and software-defined networking, but we will also cover other topics.

At the same time, it is a follow-up to the Bachelor lecture Computer Networks (and serves, in this sense, as a Computer Networks II). We shall delve more deeply into topics that were only superficially discussed in Computer Networks.

Given that this is highly research-focussed and tries to be up-to-date, this lecture is going to be a learning experience for everybody involved, in particular, the lecturer himself! Do not expect ultimate answers, rather expect questions. Also, to profit most from this lecture, it makes a lot of sense to read some of the research papers yourself (in this sense, the lecture has some aspects of a seminar as well).

Teaching language is English.

Literaturangaben

- **W. Stallings:** *Foundations of Modern Networking*, Addison Wesley

Additional material to be distributed via a Mendeley group Future Internet.

Verschiedenes

Hörerkreis:

Master

nächster Wiederholungstermin:

SS 2017

Modulzugehörigkeit:

Module ESS 3.1 und ESS 3.3

Homepage:

<http://www-old.cs.uni-paderborn.de/fachgebiete/fachgebiet-rechnernetze/lehre.html>

Intelligenz in eingebetteten Systemen

Dozent: Kleinjohann, Bernd und Kleinjohann, Lisa

Büro: FU 214

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die steigende Verfügbarkeit unterschiedlichster Sensoren und Aktoren in technischen Systemen erlaubt die Realisierung zunehmend komplexerer, intelligenter Funktionalität mit Hilfe eingebetteter Software. In dieser Vorlesung werden Probleme und Ansätze/Verfahren für die Realisierung solcher intelligenten Funktionsweisen in eingebetteten Systemen vorgestellt. Als Beispiele dienen eingebettete Systeme aus verschiedenen Anwendungsgebieten wie dem Automobilbereich, der Telekommunikation oder der Robotik.

Folgende Gebiete werden behandelt:

Anwendungsszenarien und Architekturen

Bildverarbeitung

Sensorfusion

Karten und Navigation

reaktive Agenten / verhaltensbasierte Programmierung, affective computing

Planen und Grundlagen kooperativen Handelns

Lernen (verstärkendes Lernen, Bayessches Lernen)

Wesentliche Gesichtspunkte sind Realzeitfähigkeit und Ressourcenbeschränkungen der Verfahren als Randbedingungen für eine eingebettete Realisierung

Literaturangaben

Folien werden ins Netz gestellt

- **St. Russel, P. Norvig:** *Artificial Intelligence: A Modern Approach*, Pearson 2010
- **N. Nilsson:** *Artificial Intelligence: A New Synthesis*, Morgan Kaufman Publishers 1998
- **R. Arkin:** *Behavior-Based Robotics*, MIT Press, 1998

Verschiedenes

Prüfungsgebiet:

Informatik: ESS, Modul III.3.4 u. III.3.6;
Computer Engineering: Modul Real-time/Embedded Systems

Scheinerwerb:

mündliche Prüfung

Studiennachweis (Lehramt):

mündliche Prüfung

nächster Wiederholungstermin:

voraussichtlich SS 2017

weiterführende Veranstaltungen:

Projektgruppe

Homepage:

Unterlagen etc. siehe Paul

Metaheuristics for Hardware Evolution (In English)

Dozent: Kaufmann

Büro: O3.116

Sprechstunde: nach Absprache

Inhaltsangabe

This lecture introduces modern metaheuristics, a family of optimization algorithms based on the principles of analogy, induction and decomposition and inspired by mechanisms such as the annealing process in metallurgy and the biological evolution of species. Metaheuristics are used when computational challenges are getting to large and complex to solve them optimally. Examples for such challenges are the design of digital and analog circuits, the identification of bugs in programs, finding new types of antennas, and the computation of large scheduling plans.

While the lecture focuses on the algorithmic aspects of metaheuristics and on statistical tools for algorithm evaluation and comparison, the labs are organized as theoretical and programming exercises requiring the students to implement, apply, evaluate, and compare the algorithms introduced in the lecture on tasks such as floorplanning, chip heat flow forecasting, evolution of hardware signal classifiers, optimization of smart grid extensions, and electric power grid restoration.

Literaturangaben

- **Weicker, Karsten:** *Evolutionäre Algorithmen*, Springer, 2007. ISBN 978-3-8351-9203-4
- **Kruse et al.:** *Computational Intelligence - A Methodological Introduction*, Springer, 2013. ISBN 978-1-4471-5012-1
- **Kruse et al.:** *Computational Intelligence [DE]*, Vieweg+Teubner-Verlag, Wiesbaden, 2011. ISBN 978-3-8348-1275-9
- **Wang et al.:** *Electronic Design Automation*, Morgan Kaufmann, 2009. ISBN: 0-1237-4364-8

Verschiedenes

Hörerkreis:

Informatik Master / Computer Engineering
Master

Modulzugehörigkeit:

ESS

vorausgesetzte Kenntnisse:

Grundlagen der technischen Informatik / Digitaltechnik

nützliche Parallelveranstaltungen:

Eingebettete Prozessoren

weiterführende Veranstaltungen:

Hardware/Software Codesign

nächster Wiederholungstermin:

SS17

Homepage:

<https://sites.google.com/site/paulkaufmann/lectures>

Network Simulation

Dozent: Dressler

Büro: F1.401

Sprechstunde: nach Absprache

Inhaltsangabe

This course offers the chance to get in touch with the network simulation framework OMNeT++. Simulation is one possibility (others are experiments or mathematical analysis) to evaluate the performance of systems which might even not be available in reality.

After getting a basic introduction to simulation and modeling, we will study a small example project already using OMNeT++. Finally, we will work in small groups of 2-3 people on interesting research-focused projects of the areas vehicular communication and wireless sensor networks.

Literaturangaben

- **Averill M. Law:** *Simulation, Modeling and Analysis*, Singapore, McGraw-Hill, 2007.

Verschiedenes

Hörerkreis:

Master Informatik und Computer Engineering

Homepage:

<http://www.ccs-labs.org/teaching/nwsim/>

Swarm Robotics (in English)

Dozent: Hamann

Büro: F1.221

Sprechstunde: (nach Vereinbarung)

Inhaltsangabe

Swarm robotics is a comparatively young field of research. Its predominant paradigm – decentral control – is inspired by biological systems such as social insects and bird flocks. These systems show complex patterns based on simple behaviors which constitutes the engineer’s dream: a little effort creates a complex system. One of the main studies in swarm robotics is to find and establish a design methodology for decentral systems (how to design and how to program robot swarms). The challenge is to bridge the gap between the globally defined task and the locally restricted behavior of a particular robot. Following this path, it is essential to understand how swarms are organized which is achieved by diverse modeling techniques.

In this lecture, fundamental concepts of swarm robotics are introduced, such as decentralized control, self-organization, and scalability. We move on by investigating typical swarm behaviors, such as aggregation, dispersion, goal searching, cooperative transport, and flocking. In order to find and understand the fundamental concepts that govern swarm behaviors, swarms are modeled. A collection of modeling techniques is introduced, such as simulations, rate equations, and stochastic processes. Several key concepts to understand swarms are presented (microscopic vs. macroscopic level, fluctuations, local sampling).

Subjects that are discussed in-depth include collective decisions and the implementation of adaptive aggregation. Collective decisions are a typical example of swarm intelligence. Based on local rules the overall system converges to a common consensus. It is also an example of interdisciplinary research connecting computer science, engineering, physics, biology, and sociology. On the basis of an adaptive aggregation behavior we draw the connecting line from biological inspiration, to behavioral models, robot algorithms, the resulting swarm behaviors, and possible design techniques.

Literaturangaben

- **Dario Floreano and Claudio Mattiussi:** *Bio-inspired Artificial Intelligence: Theories, Methods, and Technologies*, MIT Press, 2008

Verschiedenes

Hörerkreis:

Master Computer Science

Modulzugehörigkeit:

ESS III.3.2,III.3.4,III.3.6

Prüfungsform:

Oral exam

Leistungspunkte:

4 ECTS

nächster Wiederholungstermin:

SS 17

Vehicular Networking

Dozent: Dressler

Büro: F1.401

Sprechstunde: nach Absprache

Inhaltsangabe

Today's automotive industry is increasingly relying on computer science in product innovation. Young professionals are expected to have specialized knowledge in the fields of electronics, software and vehicular networks – both in-car networks and networks of moving cars. This lecture teaches important concepts from these domains, starting with in-car networks (from individual electronic control units, modern bus systems, system and network architectures, to driver assistance functions, security and safety). The lecture then moves to networks of moving cars (from communication technology and system architectures, to the design of advanced traffic information systems, security and safety). Particular emphasis is given to the relevant question of balancing users' privacy with their safety and security.

Literaturangaben

- **Sommer and Dressler:** *Vehicular Networking*, Cambridge University Press, 2014

Verschiedenes

Hörerkreis:

Master Informatik, Master Computer Engineering

nächster Wiederholungstermin:

Sommer 2017

Homepage:

<http://www.ccs-labs.org/teaching/c2x/>

Data and Information Visualization

Dozent: Domik

Büro: F2.204

Sprechstunde: Dienstags, aber bitte Webseite beachten.

Inhaltsangabe

- Definitions and history of Visualization
- Data foundations
- Preprocessing of data
- User and Tasks
- Design/Mapping from data attributes to visual variables (pictures)
- Visualization techniques, incl.
 - Visualization of 3D Scalars (e.g. volume visualization)
 - Visualization of vector fields (flow visualization)
- Interaction
- Evaluation of visualization

Assignments will include the programming of preprocessing and visualization techniques. You have to already be an efficient programmer, this course is not a course to learn programming, but rather to use such skills.

Verschiedenes

Hörerkreis:

Master Informatik

Prüfungsform:

Submit assignments and oral exam

nächster Wiederholungstermin:

Summer 2017

Homepage:

<https://www-old.cs.uni-paderborn.de/fachgebiete/ag-domik/lehre.html>

Modulzugehörigkeit:

Module III.4.1 "Computer Graphics and Visualization"

vorausgesetzte Kenntnisse:

Introduction to Computer Graphics

Vorbesprechung:

First lecture

Modelling User Interfaces

Dozent: Szwillus

Büro: F2.119

Sprechstunde: siehe Webseite

Inhaltsangabe

The class starts by dealing with the concept of modelling in a “Computer Science” way as such. Models are referred here with the following definition: “A model is an artefact, such as a drawing, a text, or a physical object supposed to reflect or represent certain relevant aspects of a real system.” Under this motto a series of different types of artefacts relevant under certain aspects are covered. This starts with task modelling, where we look into several task modelling approaches, including some tools to work with. Then we enter the area of dialogue models, which are divided into two distinct groups - the user interaction models and the control models. The two types describe “the same thing”, with different goals and results. This chapter deals with several approaches and their interconnections, if applicable by looking at corresponding tools. The class finishes with describing a model-based user interface design process.

Verschiedenes

Hörerkreis:

Master Computer Science, Business Informatics

Prüfungsform:

Oral module exam

vorausgesetzte Kenntnisse:

No special requirements

weiterführende Veranstaltungen:

Web Modelling (Module III.4.6)

Vorbesprechung:

no

Modulzugehörigkeit:

III.4.5 Entwicklung von Benutzungsschnittstellen (M.079.3425) und III.4.6 Modellbasierte Entwicklung von Benutzungsschnittstellen (M.079.3426)

Leistungspunkte:

4

nützliche Parallelveranstaltungen:

Usability Engineering Practice (Module III.4.5)

nächster Wiederholungstermin:

WS 17/18

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/fg-mci/lehre/ss-2016/modelling-user-interfaces.html>

Seminar: Algorithmen für verteilte Systeme

Dozent: Scheideler

Büro: F2.326

Sprechstunde: Do, 16-17 Uhr

Inhaltsangabe

Das Seminar wird auf aktuellen Konferenzveröffentlichungen im Bereich der Algorithmen für skalierbare und robuste verteilte Systeme aufbauen. Dabei wird ein Schwerpunkt auf Arbeiten gelegt, die wichtige theoretische Ergebnisse in diesem Bereich beigetragen haben. Themenschwerpunkte sind:

- Overlaynetze, über das Internet, die es erlauben, Millionen von Teilnehmern effektiv zu verwalten
- verteilte Speichersysteme, die gegen massive Angriffe immun sind
- verteilte Verfahren für programmierbare Materie

Literaturangaben

Literaturangaben werden auf der Webseite des Seminars zu finden sein.

Verschiedenes

Hörerkreis:

i-m

Modulzugehörigkeit:

MuA, Masterstudiengang

Prüfungsform:

Vortrag und Ausarbeitung

Leistungspunkte:

4

vorausgesetzte Kenntnisse:

Verteilte Algorithmen und Datenstrukturen

nützliche Parallelveranstaltungen:

-

weiterführende Veranstaltungen:

-

Vorbesprechung:

Siehe die Webseite

Homepage:

<http://www-old.cs.uni-paderborn.de/fachgebiete/fg-ti/lehre0/ss-2016/avads.html>

Seminar Algorithmic Game Theory (in english)

Dozent: Skopalik

Büro: F1.209

Inhaltsangabe

In algorithmic game theory we analytically study scenarios involving the interaction of rational agents, such as traffic systems with selfish agents (e.g., cars) routing through a network. Typical questions concern the incentives in such environments - will the agents agree on a common routing pattern? Is there more than one such stable outcome? Is it “fair” or “efficient”? Will the dynamics evolving from selfish behavior converge and how long does this take?

Another prominent issue are efficient algorithms for computing stable allocations in systems with selfish agents, e.g., when matching buyers to sellers, goods to consumers, or partners to each other. Such algorithms use information and preferences reported by agents, and agents might want to manipulate the algorithm by misreporting preferences to get a better good/partner/etc. The challenge here is not only to design algorithms with socially favorable outcomes, but also to manage the incentives using, e.g., payments or threats of punishment, to discourage agent manipulation.

Traditionally, techniques from microeconomics are used for mathematical formulation and solution of such problems. Recently, however, these problems are arising within computational environments - most prominently on the Internet and in e-commerce - and here ideas from game theory must be combined with approximation algorithms, distributed computing, and computational complexity.

Verschiedenes

Hörerkreis:

Informatik

Prüfungsform:

Seminar

Homepage:

<http://www.hni.uni-paderborn.de/alg/lehre>

Modulzugehörigkeit:

MuA III.2.1, III.2.2

vorausgesetzte Kenntnisse:

Lecture on Algorithmic Game Theory
Data Structures and Algorithms

Master Seminar on Selected Topics in Wireless Networks

Dozent: Dressler

Büro: F1.401

Sprechstunde: nach Absprache

Inhaltsangabe

This seminar is being offered as part of the "Master Seminar" series, each issue of which treats current topics in the context of active research domains.

Verschiedenes

Hörerkreis:

Master Informatik, Master Computer Engineering

nächster Wiederholungstermin:

Winter 2016/2017

Vorbesprechung:

erste Semesterwoche

Homepage:

<http://www.ccs-labs.org/teaching/ms/>

Projektgruppe: History in Paderborn App

Dozent: Oberthür, Engels

Büro: ZM1-03-08

Sprechstunde: Nach Absprache

Inhaltsangabe

Agile development & operation (DevOps) of the History in Paderborn App including a web based content management system.

Goal of this project group is to develop a mobile app which enables people to explore Paderborns history in different places all over the city. Content for this app will be created in seminars by students of the faculty of Arts and Humanities starting in 2015. This content shall be delivered via a web-based content management system where the supervisor can comment and release it.

The project group will develop and maintain this content management system together with the mobile app itself. The agile Scrum method combined with the DevOps life cycle will be used as development and operation method. DevOps is a concept that emphasizes the necessity that development and operation grow closer together. Result of this concept is among other things the opportunity to continuously deliver new features.

An outstanding feature of this project group is the fact that it is running steadily (and has already started). Like in the reality of modern development teams participants can join the project group continuously (limited by the maximum team size). Maintaining the knowledge in such a group and being responsible for a real used system are only two challenges of this project group.

Verschiedenes

Hörerkreis:

Master Informatik

Modulzugehörigkeit:

SWT

Homepage:

[http://is.uni-paderborn.de/
fachgebiete/fg-engels/lehre/ss15/
hip-app/pg-hip-app.html](http://is.uni-paderborn.de/fachgebiete/fg-engels/lehre/ss15/hip-app/pg-hip-app.html)

Oberseminar: Algorithmen und Komplexität

Dozent: Meyer auf der Heide

Büro: F1.301

Sprechstunde: Mi, 13-14 Uhr

Inhaltsangabe

Im Oberseminar werden Arbeiten und Themen aus dem Bereich der Fachgruppe Friedhelm Meyer auf der Heide präsentiert. Mitarbeiter stellen aktuelle Ergebnisse und Fragestellungen ihrer laufenden Arbeiten vor. Studierende halten Abschlussvorträge ihrer Studienarbeiten und Diplomarbeiten. Eingeladene Gäste präsentieren ihre Gastvorträge. Neue Originalarbeiten werden vorgestellt.

Die Studierenden sind herzlich eingeladen an den Vorträgen teilzunehmen. Sie können sich durch das Seminar einen Überblick über aktuelle Forschungsfragen der Fachgruppen verschaffen. Momentane Schwerpunkte: Theorie paralleler Systeme, Randomisierte Algorithmen, Komplexitätstheorie, Drahtlose Netzwerke, Geometrische Algorithmen, Algorithmen in der Computergrafik.

Verschiedenes

Hörerkreis:

Bachelor, Master Informatik

Prüfungsform:

kein Schein

nächster Wiederholungstermin:

Seminar findet jedes Semester statt

Homepage:

www.hni.upb.de/alg/oberseminar

Oberseminar des SFB 901

Dozent: Meyer auf der Heide

Büro: F1.301

Sprechstunde: Mi, 13-14 Uhr

Inhaltsangabe

In diesem Oberseminar berichten die am SFB 901 „On-The-Fly Computing – Individualisierte IT-Dienstleistungen in dynamischen Märkten“ beteiligten Arbeitsgruppen über aktuelle Fragestellungen und Ergebnisse ihrer laufenden Forschungsarbeiten. Die Vorträge werden entweder von Mitgliedern des SFB 901 oder von Gastwissenschaftlern gehalten. Eine Übersicht über die geplanten Termine und Vorträge lässt sich unter <http://sfb901.uni-paderborn.de> abrufen. Studierende sind herzlich eingeladen teilzunehmen, um sich einen Überblick über die Themenbereiche des SFB 901 zu verschaffen.

Verschiedenes

Hörerkreis:

Bachelor, Master Informatik

Prüfungsform:

kein Schein

nächster Wiederholungstermin:

Seminar findet jedes Semester statt

Homepage:

<http://sfb901.uni-paderborn.de>

Webbasierte Informationssysteme

Dozent: Lettmann

Büro: O4.173

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Webbasierte Informationssysteme sind aus unserem heutigen Leben und Arbeiten nicht mehr wegzudenken. Von Bereitstellung und Austausch von Informationen über Kommunikation bis zur Kooperation spielt das weltweite Datennetz eine immer wichtiger werdende Rolle. Die Vorlesung gibt einen Überblick der relevanten konzeptuellen Grundlagen Webbasierter Informationssysteme. Hierzu zählen grundsätzliche Architekturen solcher Systeme, Kommunikationsmodelle, Dokumentensprachen und spezielle Client- und Server-Technologien. Neben der Vermittlung der konzeptuellen Grundlagen in der Vorlesung sollen Technologien im Rahmen der Übungen auch praktisch eingesetzt werden.

Diese Veranstaltung ist NICHT für das Studium Generale der Informatik-Studenten gedacht.

Literaturangaben

- **Meinel/Sack:** *WWW - Kommunikation, Internetworking, Web-Technologien*, Springer, 2004, ISBN 978-3-642-18963-0
- **Meinel/Sack:** *Internetworking - Technische Grundlagen und Anwendungen*, X.media.press, 2012, ISBN 978-3-540-92939-0

Verschiedenes

Hörerkreis:

Bachelor Medienwissenschaften / Master
Sport und Gesundheit

Modulzugehörigkeit:

Modul 10 Grundkonzepte des WWW / Modul M8 Multimediale Wissensvermittlung im Gesundheitssport

Prüfungsform:

Klausur

Leistungspunkte:

5 ECTS / 6 ECTS (für M8 mit Zusatzblock)

vorausgesetzte Kenntnisse:

Einführung in webbezogene Sprachen, Einführung in die Informatik

nächster Wiederholungstermin:

SS 2017

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/fg-kleine-buening/lehre/webbasierte-informationssysteme.html>

Einführung in die Informatik für Geisteswissenschaftler

Dozent: Selke

Büro: F1.101

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Studierenden erhalten einen Einblick in die Denk- und Arbeitsweisen der Informatik. Sie lernen Basiskonzepte und -begriffe der verschiedenen Teilbereiche der Informatik kennen und erwerben ein Basisverständnis bezüglich Formalisierung sowie die Fähigkeit, Software nicht mehr von der Nutzer- sondern von der Erstellerseite her zu denken.

Die Studierenden gewinnen einen Einblick in die gesellschaftliche Bedeutung der Informatik und der digitalen Medien sowie in die Fachsprache der Informatik und damit die Fähigkeit, zwischen verschiedenen Fachsprachen und Fachkulturen zu moderieren.

Die Veranstaltung umfasst unter anderem folgende Inhalte: Von Schrift und Zahl zu Hypermedia, Grundlagen des Internet, Darstellung von Daten in Computern, Trennung von Form und Inhalt am Beispiel HTML, Formale Beschreibung von Regelsystemen, Algorithmen, Kernkonzepte höherer Programmiersprachen, Einführung in die Programmierung am Beispiel JavaScript, Abstrakte Datenstrukturen.

Verschiedenes

Hörerkreis:

Medienwissenschaften (BA), mawi, mi2, mm

Prüfungsform:

Klausur, Mini-Projekt, aktive Teilnahme an den Übungen

vorausgesetzte Kenntnisse:

keine

weiterführende Veranstaltungen:

Einführung in Web-bezogene Sprachen

Vorbesprechung:

Vorlesung in der ersten Semesterwoche

Modulzugehörigkeit:

Modul 9: Einführung in die Informatik für Geisteswissenschaftler

Leistungspunkte:

8

nützliche Parallelveranstaltungen:

keine

nächster Wiederholungstermin:

Sommer 2017

Homepage:

koala.uni-paderborn.de

5 Raum für Notizen

6 Ergebnisse der Veranstaltungskritik

Hallo,

üblicherweise findet Ihr hier an dieser Stelle eine Übersicht über die Ergebnisse der Veranstaltungskritik.

Diese können, aus datenschutzrechtlichen Gründen, nur in der gedruckten Fassung des V-Koms veröffentlicht werden. Wenn euch die Ergebnisse interessieren, könnt Ihr diese jeder Zeit bei uns im Fachschaftsbüro E1.311 ansehen.

Stundenplan

Uhrzeit	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
7 - 8					
8 - 9					
9 - 10					
10 - 11					
11 - 12					
12 - 13					
13 - 14					
14 - 15					
15 - 16					
16 - 17					
17 - 18					
18 - 19					
19 - 20					