

Universität Paderborn



Informatik
**Veranstaltungs-
Kommentar**

Für

Informatik ▷ Bachelor
▷ Master

Computer Engineering

Ingenieurinformatik

Lehramt Informatik GyGe

Für das SoSe 17

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	13
5	Raum für Notizen	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

V.i.S.d.P.: Eduard Maas

ISSN: 1868-0690

Redaktion: 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 leider 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/>

Eduard Maas, Alex Wiens

V-Kom-Redaktion für das SoSe 2017

2 Mitarbeitende der Informatik

Name	E-Mail	Telefon	Raum
Agne, Andreas	agne@upb.de	4348	O3.125
Ahlers, Ulrich, Dipl.-Math., T.	uli@hni.upb.de	6700	F2.320
Allwright, Michael	michael.allwright@uni-paderborn.de	3320	ZM1.02-10
Anjorin, Anthony	anthony.anjorin@upb.de	5465-224	ZM1.03-06
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
Blobel, Johannes	blobel@ccs-labs.org	6494	F U.407
Blömer, Johannes	bloemer@upb.de	6651	F2.101
Blößl, Bastian	bloessl@ccs-labs.org	6494	F U.407
Blume, Bodo	bodo.blume@uni-paderborn.de	1750	O3.173
Bobolz, Jan	jan.bobolz@upb.de	6654	F2.111
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
Buse, Dominik	buse@ccs-labs.org	6560	F1.416
Dann, Andreas	andreas.dann@upb.de	5719	ZM1.02-07
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
Engels, Gregor	engels@upb.de	3337	ZM1.03.45
Feldkord, Björn	bjoernf@mail.uni-paderborn.de	6434	F1.125
Feldmann, Michael			
Feldmann, Rainer.	obelix@upb.de	6720	F2.401
Feldotto, Matthias	feldi@mail.upb.de	6452	F1.203
Fischer, Matthias, Dr. rer. nat.	mafi@upb.de	6466	F1.223
Geismann, Johannes	johannes.geismann@upb.de	3309	ZM1.02-11
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@uni-paderborn.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

Name	E-Mail	Telefon	Raum
Haupt, Jutta	jutta@upb.de	3312	ZM1.02.08
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
Hucke, Marion	marion.hucke@upb.de	6695	F2.411
Hüllermeier, Eyke	eyke@upb.de	1771	O4.258
Hülsmann, Adrian	klecks@upb.de	6623	F2.124
Jahn, Claudia	claudia.jahn@uni-paderborn.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.uni-paderborn.de	6433	F1.125
Karl, Holger	hkarl@mail.uni-paderborn.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@uni-paderborn.de	6411	F1.428
Kleine Büning, Hans	kbcs1@upb.de	3360	O4.255
Klingler, Florian	klingler@ccs-labs.org	6493	F1.319
Kolb, Christina	ckolb@mail.upb.de	6725	F2.406
König, Jürgen ab 15.07.			
Kouchaksaraei Hadi Razzaghi	hadi.razzaghi.kouchaksaraei@uni-paderborn.de	1758	O3.170
Krämer, Julia-Desirée	juliadk@mail.upb.de	5388	O4.131
Kraus, Hans-Joachim	hajo.kraus@upb.de	6538	F1.410
Kreiss, Lydia	Lydia.Kreiss@uni-paderborn.de	6501	F1.404
Krohn, Jörg-Peter	krohn@upb.de	3325	E1.103
Krüger, Jessica	jessica.krueger@upb.de		F2.114
Krüger, Stefan	stefan.krueger@upb.de	3309	ZM1.02 -12
Lange, Christina	chlange@mail.upb.de	1749	O4.213
Lefèvre, Jonas	jonas.levevre@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.uni-paderborn.de	6427	F1.119
Magenheim, Johann S.	jsm@uni-paderborn.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	memedi@ccss.labs.org	6513	F1.416
Meyer auf der Heide, F.	fmadh@upb.de	6480	F1.301
Meyer, Christina	chmeyer@upb.de	6733	E1.106/F2.416

Name	E-Mail	Telefon	Raum
Mohammadi, Hassan ab 01.08.			
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@uni-paderborn.de	6600	F2.416
Nabeel, Muhammad	nabeel@ccs-labs.org	6515	F 1.412
Neugebauer, Jonas	jonas.neugebauer@uni-paderborn.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@uni-paderborn.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
Riechers, Sören	soerenri@mail.upb.de	6427	F1.119
Roger, Irene	irene.roger@upb.de	6620	F2.122
Schäfer, Johannes	jschaefer@mail.upb.de	6428	
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@uni-paderborn.de	2306	ZM1.02-14
Schubert, Philipp	phillip.schubert@upb.de	5584	ZM1.02-10
Schulte, Carsten	carsten.schulte@upb.de	6343	F2.116
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
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	F2.323
Szwillus, Gerd	szwillus@upb.de	6624	F2.119
Tetz, Alexander	alexandwer.tetz@upb.de	5465-215	ZM1.03-10
Thissen, Thomas	tici@upb.de	6701	F2.320
Töws, Manuel	mtöws@upb.de	4272	O4.122
Travkin, Oleg	oleg82@upb.de	1709	O4.131
Utermöhle, Michael	mike@upb.de	6666	F2.224

Name	E-Mail	Telefon	Raum
Wahby, Mostafa	mostafa.wahby@uni-paderborn.de	6468	F1.213
Walther, Sven	sven.walther@upb.de	3763	O4.122
Wassing, Heinz-Georg,	koala@upb.de	6430	F1.122
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
Yevgen, Mexin	yevgen@mail.upb.de		F2.416
Zahn, Gero	gero.zahn@upb.de	6519	F1.107
Gast, Aulon Shabani			

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 Cam- pus	IRB-Support@upb.de	3322	E1.303
Rechnerbetreuung Informatik Fürs- tenallee	IRB-Support@upb.de	6755	F2.521
Rechnerbetreuung Informatik Tech- niker	IRB-Support@upb.de	3318/4332	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

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

Hauptstudium Bachelorstudiengang

Brunthaler	Compilerbau	(1.1)	--
Platenius	Modellbasierte Softwareentwicklung	(1.1)	20
Wehrheim	Softwaremodellierung mit formalen Methoden	(1.1)	21
Feldmann	Optimization	(2.1)	22
Scheideler	Verteilte Algorithmen und Datenstrukturen	(2.1)	23
Schroeder	Methoden des Algorithmenentwurfs	(2.?)	24
Platzner	Eingebettete Prozessoren	(3.1)	--
Szwillus	Gestaltung von Webauftritten	(4.1)	--

Proseminare

Wehrheim	Software Analyse Verfahren und Werkzeuge		25
Fischer & Schroeder	Algorithmische Geometrie		26
Jäger	Digitale Signaturen		--
Karl	Rechnernetze		27
Dressler	Verteilte eingebettete Systeme		--
Böttcher	Neuere IT-Technologie Trends		--

Brunthaler	Optimierungstechniken für Dynamisch-typisierte Programmiersprachen	--
Becker	Modellbasierte Softwareentwicklung	28

Masterstudiengang

Anjorin	Software Quality Assurance	(1.1,1.5,1.6)	29
Brunthaler	Language based Security	(1.3)	--
Bodden	Designing Code Analyses for Large-Scale Software Systems	(1.1,1.5)	30
Bodden	Build It, Break It, Fix It	(1.5)	32
Busa-Fekete	Reinforcement Learning	(1.4)	--
Böttcher	Databases and information systems	(1.3,1.6)	--
Busa-Fekete	Online learning and optimization	(1.4)	--
Skopalik	Online Algorithmen	(2.1,2.2)	--
Fischer	Computational Geometry	(2.1,2.2)	33
Blömer	Cryptographic protocols	(2.3)	34
Blömer	Cryptography - Provable Security	(2.3)	35
Blömer	Re(AC) ^t :Reputation and Anonymous	(2.3)	36
Plessl	Hardware/Software Codesign	(3.4,3.5,3.6)	--
Jager	Bitcoins und Cryptowährungen	(3.1,3.3)	--
Dressler	Vehicular Networking	(3.1,3.3,3.6)	38
Kaufmann	Adaptive Hardware and Systems	(3.4,3.5)	39
Kleinjohann & Kleinjohann	Intelligenz in eingebetteten Systemen	(3.4,3.6)	40
Karl	Empirical Performance Evaluation	(3.1,3.3)	41
Simon	Architektur paralleler Systeme	(3.?)	--
Mohammadi	Algorithms for Synthesis and Optimization of Integrated Circuits	(3.4,3.5,3.6)	42
Karl	Future Internet	(3.1,3.3)	43
Szwillus	Web Modelling	(4.5,4.6)	--
Domik	Data and Information Virtualization	(4.1)	44

Seminare

Blömer	Algorithmic Complexity	(2.1,2.2,2.3)	45
Jäger	Aktuelle Themen der Computersicherheit	(2.3,3.1,3.2,3.3)	--
Scheideler	PRISMA	(2.1,2.2)	--
Karl	Computer Networks	(3.1,3.3)	--
Kaufmann	Binary Decision Diagrams	(3.4,3.5)	--
Jäger	Aktuelle Themen der Computersicherheit	(2.3,3.1,3.2,3.3)	--
Dressler	Selected Topics in Wireless Networking	(3.1,3.3)	46

Projektgruppen

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

Kaufmann	Biochips		--
Anjorin	Designing a Library for Model Analyses using MPS		--
Plessl	OpenCL and the 7 Dwarfs		--
Engels & Oberthür	History in Paderborn App 5		47
Bodden	Secure Integration of Cryptographic Software		48
Scheideler	Trusted Communication Modules		49

Oberseminare

Brunthaler	Compilerbau und Programmiersprachen	(SWT&IS)	--
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)	50
Blömer	Codes & Kryptographie	(MuA)	--
Domik	Interactive 3D Graphics and Games	(MuA/MMWW)	--
Meyer auf der Heide	SFB 901 On-The-Fly Computing	(MuA)	51
Scheideler	Theoretische Informatik	(MuA)	52
Platzner	Approximate Computing	(ESS)	--

Jäger	Computersicherheit	(ESS)	--
Dressler	Distributed Embedded Systems	(ESS)	--
Plessl	Hochleistungs-IT-Systeme	(ESS)	--
Karl	Rechnernetze	(ESS)	--
Platzner	Technische Informatik	(ESS)	--
Keil	Informatik und Gesellschaft	(MMWW)	--
Domik	Computergrafik, Visualisierung und Bildverarbeitung	(MMWW)	--
Szwillus	Mensch-Computer-Interaktion	(MMWW)	--

Didaktik der Informatik

Schulte	Fachdidaktische Grundlagen		53
---------	----------------------------	--	----

Lehrveranstaltungen für andere Studiengänge

Selke	Einführung in die Informatik für Geisteswissenschaftler		54
Lettmann	Webbasierte Informationssysteme		55
Selke	Gesellschaft und Informationstechnik		56

Allgemeine Veranstaltungen

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

4.2 Informatik

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 grafische Programmierung mit Swing

Verschiedenes

Hörerkreis:

Bachelor Informatik, 2. Semester

vorausgesetzte Kenntnisse:

GP1

Prüfungsform:

Klausur

Homepage:

<http://cs.uni-paderborn.de/sms/teaching/>

Softwaretechnikpraktikum

Dozent: Bodden

Büro: ZM1.02-09

Sprechstunde: Nach Absprache

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, Ingenieurinformatik
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.

nächster Wiederholungstermin:

Sommersemester 2018

weiterführende Veranstaltungen:

Modellbasierte Softwareentwicklung

Vorbesprechung:

erster Vorlesungstermin

Homepage:

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

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 Graphenalgorithmen 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

vorausgesetzte Kenntnisse:

keine

nächster Wiederholungstermin:

SS 2018

Modulzugehörigkeit:

Info 1. Studienabschnitt, MUA

Leistungspunkte:

8 ECTS

weiterführende Veranstaltungen:

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

Homepage:

<http://cs.uni-paderborn.de/ti/lehre/veranstaltungen/ss-2017/datenstrukturen-und-algorithmen/>

Grundlagen der Technischen Informatik (GTI)

Dozent: Hellebrand

Büro: P1.6.08.1

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Lehrveranstaltung vermittelt Grundlagen des Entwurfs digitaler Schaltungen und stellt Verfahren zur Optimierung von Schaltnetzen und Schaltwerken vor. Die Vorlesung behandelt folgende Themen:

- Darstellung von Information
- Boolesche Algebra
- Gatter und Schaltnetze
- Logikoptimierung
- Automaten
- Schaltwerke
- Arithmetische Einheiten als Entwurfsbeispiele
- Entwurf auf Register-Transfer-Ebene
- Entwurf mit VHDL

In den Übungen wird der Vorlesungsstoff durch die Ausarbeitung von Beispielen vertieft. Im Praktikum werden durch das selbständige Arbeiten in Gruppen zu 1-2 Personen Erfahrungen mit einer modernen Hardware- Entwurfsumgebung gesammelt und praktische Fertigkeiten in VHDL erworben.

Literaturangaben

- **J. P. Hayes:** *Introduction to Digital Logic Design*, Addison-Wesley, 1993, ISBN 0-201-15461-7
- **J. F. Wakerly:** *Digital Design*, 4th Edition, Upper Saddle River, NJ: Pearson / Prentice Hall, 2007

Verschiedenes

Hörerkreis:

Pflichtveranstaltung im Bachelor Informatik,
Computer Engineering und Elektrotechnik

Prüfungsform:

Klausur

vorausgesetzte Kenntnisse:

keine

nächster Wiederholungstermin:

SS 2018

Vorbesprechung:

Es wird alles in der 1. Vorlesungsstunde erklärt.

Homepage:

<http://www.date.upb.de/pages/de/lehre.php>

Konzepte und Methoden der Systemsoftware

Dozent: Dressler

Büro: F1.401

Sprechstunde: nach Absprache

Inhaltsangabe

Betrachtet man Lehrinhalte klassischer Teilgebiete der Informatik wie Übersetzerbau, Betriebssysteme, Datenbanksysteme, Rechnernetze, Verteilte Systeme oder Rechnerarchitektur, so kann man feststellen, dass es immer wieder fundamentale Problemstellungen gibt, die in den einzelnen Gebieten als Varianten auftauchen und dort mit entsprechenden Verfahren gelöst werden. Es liegt daher nahe, diese Einzelphänomene aus ihrem Kontext herauszulösen, ihre Gemeinsamkeiten herauszuarbeiten und sie als allgemeine Phänomene einmalig und grundlegend zu behandeln.

Ziel der Vorlesung ist die Vermittlung allgemeiner Prinzipien, Konzepte, Methoden und Techniken, wie sie in komplexen HW/SW-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

- **Stallings, William:** *Operating Systems: Internals and Design Principles*, 8th ed. Upper Saddle River, Pearson Prentice Hall, 2014
- **Tanenbaum, Andrew S. and Wetherall, David J.:** *Computer Networks*, 5th ed., Prentice Hall, 2011

Verschiedenes

Hörerkreis:

Bachelor Informatik

nächster Wiederholungstermin:

Sommer 2018

Modulzugehörigkeit:

ESS

Homepage:

<http://www.ccs-labs.org/teaching/kms/2017s/>

Grundlagen Mensch-Maschine Wechselwirkung

Dozent: Keil

Büro: F1.428

Sprechstunde: n.V.

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 18

Homepage:

koala.uni-paderborn.de

Modellbasierte Softwareentwicklung (MBSE)

Dozent: Platenius

Büro: ZM1.02-13

Sprechstunde: auf Anfrage

Inhaltsangabe

Die Studierenden sollen grundlegende Verfahren zur Konstruktion großer Softwaresysteme kennen und ihre Anwendung beherrschen. Sie sollen die Vor- und Nachteile von Spezifikationstechniken erfahren, die Notwendigkeit von Design erkennen und Modelle zur Verbesserung der Softwarequalität einsetzen können. Unter anderem wird auf das Paradigma der „Model Driven Development“ eingegangen, das einen wesentlichen Produktivitäts- und Qualitätsgewinn bei der Softwareentwicklung verspricht.

Inhalt (unter anderem):

- DSL-Engineering
- Metamodellierung
- Code-Generierung
- Modelltransformationen
- Modellgetriebene Entwicklungsprozesse
- aktuelle Trends und Forschungsthemen

Verschiedenes

Hörerkreis:

Bachelor Informatik

Prüfungsform:

SWT

vorausgesetzte Kenntnisse:

Grundlagen der Programmierung (1+2),
Softwareentwurf

nächster Wiederholungstermin:

kontinuierlich

Homepage:

[https://www.hni.uni-paderborn.de/
swt/lehre/](https://www.hni.uni-paderborn.de/swt/lehre/)

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 Sprachen zur Beschreibung von zeitlichen Aspekten (Timed Automata) und gegebenenfalls auch mit zustandsbasierten Formalismen zur Spezifikationen von Daten und Operationen (Z und Object-Z) beschäftigen.

Verschiedenes

Hörerkreis:

Bachelor Informatik, 2. Studienabschnitt

Prüfungsform:

Klausur oder mündliche Prüfung

nächster Wiederholungstermin:

SS 2018

Modulzugehörigkeit:

SWT&IS

vorausgesetzte Kenntnisse:

Prädikatenlogik, etwas Lineare Algebra, formale Beweise

Homepage:

<http://cs.uni-paderborn.de/sms/teaching/>

Optimization

Dozent: Feldmann

Büro: F2.401

Sprechstunde: nach Vereinbarung

Inhaltsangabe

We present fundamentals of linear optimization. Particularly, we consider

- Introduction to Linear Optimization
- Introduction to the LP solver Cplex
- Modelling Graph Problems as Linear Programs
- The Primal Simplex Algorithm
- The Simplex Tableau
- Duality
- The Dual Simplex Algorithm
- The Network Simplex Algorithm
- The Primal-Dual Algorithm

The lecture is given in English.

Literaturangaben

- **Vasek Chvatal:** *Linear Programming*, Freeman 1983, TVX 2806
- **Dimitris Alevras, Manfred W. Padberg:** *Linear Optimization and Extensions*, Springer 2001, TLG 2186
- **Alexander Schrijver:** *Theory of Linear and Integer Optimization*, Wiley, 1999, TLG 1627
- **Robert J. Vanderbei:** *Linear Programming*, Kluwer 2001, TLG 2102
- **Jiri Matousek, Bernd Gärtner:** *Undertsanding and Using Linear Programming*, Springer 2007, TVX 3577
- **Kurt Marti, Detlef Gröger:** *Einführung in die lineare und nichtlineare Optimierung*, Physica 2000, TLG 2144

Verschiedenes

Hörerkreis:

Bachelor Informatik, 2.Studienabschnitt, ma
und LGym ma: Angewandte Mathematik,
LGym inf: Mathematische Methoden, winfo:
E2731

Prüfungsform:

Klausur

vorausgesetzte Kenntnisse:

Knowledge in linear algebra helps

Modulzugehörigkeit:

Modelle und Algorithmen, Modul II.2.1.

Leistungspunkte:

4

nützliche Parallelveranstaltungen:

Fundamental Algorithms

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
(verteilt 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

vorausgesetzte Kenntnisse:

Datenstrukturen und Algorithmen

nächster Wiederholungstermin:

SS 2018

Modulzugehörigkeit:

Info 2. Studienabschnitt, MuA

Leistungspunkte:

4 ECTS

weiterführende Veranstaltungen:

Fortgeschrittene verteilte Algorithmen und Datenstrukturen

Homepage:

<http://cs.uni-paderborn.de/ti/lehre/veranstaltungen/ss-2017/verteilte-algorithmen-und-datenstrukturen/>

Methoden des Algorithmenentwurfs

Dozent: Schroeder

Büro: F2.409

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

Hörerkreis:

Informatik Bachelor, Mathematik Master

Modulzugehörigkeit:

2. Studienabschnitt des BSc-Studiengangs,
Modul II 2.1

Prüfungsform:

mündliche Prüfung

vorausgesetzte Kenntnisse:

Vorlesung Grundlegende Algorithmen

weiterführende Veranstaltungen:

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

Proseminar Software-Analyse Verfahren und Werkzeuge

Dozent: Wehrheim

Büro: O4.225

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Programmanalyseverfahren versuchen möglichst viele Informationen über ein Programm statisch, d.h. durch Inspektion des Programmcodes, zu ermitteln. Solche Informationen werden z.B. vom Compiler zur Übersetzungszeit zur Optimierung benötigt, können aber auch Aussagen über Korrektheitseigenschaften von Programmen machen bzw. helfen einem Programmierer bei der Erstellung eines fehlerfreien Programmes.

In dem Proseminar wollen wir uns einen Überblick über verschiedene Programmanalyseverfahren und insbesondere Werkzeuge verschaffen.

Verschiedenes

Hörerkreis:

inf-b5

Prüfungsform:

Vortrag, aktive Beteiligung, schriftliche Ausarbeitung

Vorbesprechung:

erscheint auf Webseite

Modulzugehörigkeit:

übergreifend

vorausgesetzte Kenntnisse:

Programmierkenntnisse
Modellierung

Homepage:

<http://cs.uni-paderborn.de/sms/teaching/>

Proseminar: Algorithmische Geometrie

Dozent: Fischer

Büro: F1.223

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Im Proseminar werden Vorträge zu Methoden aus der Algorithmischen Geometrie angeboten. Die Vorträge werden aus unterschiedlichen Büchern zusammengestellt.

Das de Berg Buch ist von Rechnern der Uni Paderborn aus zugreifbar.

Es werden unter anderem behandelt:

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

Dozenten:

Matthias Fischer, Ulf-Peter Schroeder

Literaturangaben

- **Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars:** *Computational Geometry*, Springer-Verlag, 2008, <https://dx.doi.org/10.1007/978-3-540-77974-2>
- **Michael I. Shamos, Franco P. Preparata:** *Computational Geometry: An Introduction*, Springer-Verlag, 1985, <https://dx.doi.org/10.1007/978-1-4612-1098-6>

Verschiedenes

Hörerkreis:

Bachelor Informatik

Prüfungsform:

Seminarvortrag und Ausarbeitung

Vorbesprechung:

am ersten Veranstaltungstag

Modulzugehörigkeit:

2. Studienabschnitt des BSc-Studiengangs

vorausgesetzte Kenntnisse:

Datenstrukturen und Algorithmen ist von Vorteil

Homepage:

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

Rechnernetze (Seminar/Proseminar)

Dozent: Karl

Büro: O3.158

Sprechstunde: Montag, 13 - 14 Uhr

Inhaltsangabe

Ziel eines Proseminares ist es, die Teilnehmer mit der „seminaristischen Arbeitsweise“ vertraut zu machen. Dies beinhaltet insbesondere

- Das selbständige Erarbeiten von Sachverhalten anhand von Originalliteratur
- Das Auffinden solcher Literatur anhand von Querverweisen, eigener Recherche, etc.
- Das Selektieren wichtiger Sachverhalte und Trennen von unwesentlichen Aspekten
- Das Aufbereiten in einer adäquaten Form in schriftlicher Ausarbeitung (insbesondere die Darstellung eines zu lösenden Problems, der Lösungsidee(n), der Details einer Lösung, des Vergleichs und der kritischen Auseinandersetzung mit unterschiedlichen Lösungen)
- Das Darstellen in einem Vortrag (inklusive der Vorbereitung eines solchen Vortrages)

Diese Ziele werden in diesem Proseminar anhand von Themen der Rechnernetze verfolgt. Jeder Teilnehmer erhält ein eigenes Thema zugewiesen, das eigenständige in Ausarbeitung und Vortrag darzustellen ist (samt der dazu notwendigen Recherche und Vorbereitung).

Seminare verfolgen grundsätzlich das gleiche Ziel, finden aber auf einem thematisch etwas höheren Niveau statt (Master-Veranstaltung statt Bachelor-Veranstaltung). Wir führen Proseminare und Seminare in der Regel in einer gemeinsamen Veranstaltung durch, in der sich der Schwierigkeitsgrad insbes. durch Anspruch an Tiefe des Themas und der Literatursuche ausdifferenziert.

Verschiedenes

Hörerkreis:

Bachelor, Master

Modulzugehörigkeit:

ESS

nächster Wiederholungstermin:

SS 2018

Homepage:

<https://cs.uni-paderborn.de/cn/teaching/courses/>

Proseminar: Modellbasierte Softwareentwicklung

Dozent: Becker

Büro: ZM1.02-47

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
Vorlesungen: Softwareentwurf, Grundlagen der Programmierung
Empfohlen: Vorlesung Modellbasierte Softwareentwicklung

Vorbesprechung:

erste Semesterwoche; Termin wird noch bekannt gegeben

Homepage:

<http://www.hni.upb.de/swt/lehre>

Software Quality Assurance

Dozent: Anjorin

Büro: ZM1.03-06

Sprechstunde: Mo, 10-11 Uhr

Inhaltsangabe

The aim of the lecture is to cover approaches, technologies and strategies related to quality assurance for software systems. These include on the one hand constructive approaches such as design patterns, domain-specific languages and model driven development, and on the other hand analytic approaches such as static and dynamic testing, and monitoring. Furthermore, approaches for the improvement of the software development process and international standards like ISO 9001, 9126, CMM etc. will be covered.

Verschiedenes

Hörerkreis:

Master of Computer Science

Prüfungsform:

Oral exam

nützliche Parallelveranstaltungen:

Lectures of the modules “Analytical Methods in Software Engineering”, “Constructive Methods in Software Engineering”

Homepage:

cs.uni-paderborn.de/dbis/lehre/veranstaltungen/ss-2017/software-quality-assurance/

Modulzugehörigkeit:

SWT & IS, Modules III.1.1, III.1.5, III.1.6

Leistungspunkte:

4

nächster Wiederholungstermin:

SS 2017

Designing code analyses for large-scale software systems

Dozent: Bodden

Büro: ZM1.02-09

Sprechstunde: Nach Absprache

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:

- Intra-prozedurale Datenflussanalyse
- Algorithmen zur Konstruktion von Callgraphen
- Kontext-insensitive inter-procedurale Datenflussanalyse
- Kontext-Sensitivität mittels des Call-Strings-Ansatzes
- Value-basierte Kontexte
- Kontext-Sensitivität mittels des Funktionalen Ansatzes
- Distributive Probleme effizient mit den IFDS- und IDE-Frameworks lösen
- 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.

Homepage:

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

Modulzugehörigkeit:

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

Leistungspunkte:

4

nächster Wiederholungstermin:

Sommersemester 2017

Build It, Break It, Fix It

Dozent: Bodden

Büro: ZM1.02-09

Sprechstunde: Nach Absprache

Inhaltsangabe

This course aims on learning basic principles of secure software development in a very practical fashion.

Students will be asked to gather in teams and develop small software projects based on a formal specification, also including security requirements. In a second iteration, the developed software will be exchanged between development teams to break the implementation, i.e., find and exploit security vulnerabilities in code of other teams. Afterwards, teams will get the chance to fix found vulnerabilities and, hence, render their software product more secure.

The course will contain a smaller theoretical part in which basic strategies of secure software development are presented. Nevertheless, the course is generally of a very practical nature and since securing a software product, as well as breaking it, demands a wide variety of skills and creativity, a quite high amount of motivation and self organization is required.

Literaturangaben

- Ruef, Andrew and Hicks, Michael and Parker, James and Levin, Dave and Mazurek, Michelle L. and Mardziel, Piotr: *Build It, Break It, Fix It: Contesting Secure Development*, CCS '16

Verschiedenes

Hörerkreis:

Informatik Master

Prüfungsgebiet:

III.1.5 Analytische Methoden des Software Engineering

Vorausgesetzte Kenntnisse:

A mature understanding of the Java programming language and software security requirements. Secure software development practices will be helpful.

nächster Wiederholungstermin:

SoSe 2018

nützliche Parallelveranstaltungen:

Secure Software Engineering

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.

de Berg's, Matousek, and Klein's book can be accessed free of charge from the university's computers.

Literaturangaben

- **Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars:** *Computational Geometry*, Springer-Verlag, 2008, <https://dx.doi.org/10.1007/978-3-540-77974-2>
- **Jiri Matousek:** *Lectures on Discrete Geomtetry*, Springer-Verlag, 2002, <https://dx.doi.org/10.1007/978-1-4613-0039-7>
- **Michael I. Shamos, Franco P. Preparata:** *Computational Geometry: An Introduction*, Springer-Verlag, 1985, <https://dx.doi.org/10.1007/978-1-4612-1098-6>
- **Rolf Klein:** *Algorithmische Geometrie*, Springer-Verlag, 2005, <https://dx.doi.org/10.1007/3-540-27619-X>

Verschiedenes

Hörerkreis:
Informatik Master

Prüfungsform:
oral examination

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

Modulzugehörigkeit:
III.2.1, III.2.2

vorausgesetzte Kenntnisse:
Advanced understanding of data structures and algorithms. For example the lecture Data Structures and Algorithms of Bachelor's studies.

Cryptographic Protocols

Dozent: Blömer

Büro: F2.101

Sprechstunde: Mi, 13 - 14 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:

SS2018

Modulzugehörigkeit:

Modul III.2.2 und III.2.3

Leistungspunkte:

4

nützliche Parallelveranstaltungen:

Cryptography: Provable Security

Cryptography: Provable Security

Dozent: Blömer

Büro: F2.101

Sprechstunde: Mi, 13 - 14 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:

SS2018

Modulzugehörigkeit:

Modul III.2.2 und III.2.3

Leistungspunkte:

4

nützliche Parallelveranstaltungen:

Cryptographic Protocols

Re(AC)^t: Reputation and Anonymous

Dozent: Blömer

Büro: F2.101

Sprechstunde: Mi, 13 - 14 Uhr

Inhaltsangabe

Usually, authentication (e.g., via user name and password) serves the purpose of establishing the exact identity of someone requesting access to something (e.g., access to credit card details). However, in many scenarios access should be granted to a *group* of people and it does not matter which member of the group requests access.

For example, this may be the case for physical doors or for online subscription-based content. In these scenarios, it becomes interesting to look at more complex authentication schemes, where a user is even able to authenticate herself *anonymously*, i.e. as part of a larger group *without* revealing her own identity.

Anonymous Credential Systems offer such functionality: some party issues *credentials* to users. The credentials may be used as a means of authentication.

To authenticate, a user runs a cryptographic interactive (“zero-knowledge”) protocol where the only information revealed about her is possession of a credential (i.e. membership in the group of people with similar credentials).

In essence, this allows users to access restricted services without revealing personal information. This serves as protection against tracing a user’s activity among multiple services and creating extensive profiles that violate our expectation of privacy.

In the project group, the main goal is to theoretically examine and practically implement an anonymous credential system (and a similar treatment for anonymous reputation systems compatible with the implemented credential system).

What we offer:

- A meaningful project: We plan to publish the resulting software. Furthermore, it will be used for the SFB 901 testbed implementation of on-the-fly markets.
- Challenging theoretical tasks in the field of cryptography: design and security proof of a complex scheme built upon digital signatures, zero-knowledge proof of knowledge protocols, secret sharing schemes, and other powerful primitives.
- Challenging implementation tasks: design of nontrivial protocols and interfaces, implementation of security-critical systems, creating software support for powerful cryptographic primitives (e.g., implementing a compiler for declaratively defined cryptographic protocols), ...

As such, the project comprises interesting aspects for both the theoretically inclined and the students who enjoy working on implementation.

Note that we expect every student to understand and work on the basic theoretical aspects of the implemented schemes.

However, we explicitly encourage each participant to set a focus on theory, implementation, or a combination of those, depending on their inclination.

Literaturangaben

- **David Pointcheval and Olivier Sanders:** *Short randomizable signatures*, In *Topics in Cryptology - CT-RSA*, pages 111–126. Springer, 2016

Verschiedenes

Hörerkreis:
Informatik

Modulzugehörigkeit:
MuA

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 2018

Homepage:

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

Adaptive Hardware and Systems

Dozent: Kaufmann

Büro: O3.116

Sprechstunde: nach Absprache

Inhaltsangabe

Adaptation reflects the capability of a system to maintain or improve its performance in the context of internal or external changes, changes in the operational environment, incidental or intentional interference, or trade-offs between performance requirements and available resources.

This lecture focuses on adaptive hardware systems. After a short introduction to analog and digital reconfigurable hardware the lecture focuses on algorithms from the Computational Intelligence (CI) domain for the implementation of the adaptation and optimization mechanisms. For a detailed lecture outline please visit the teaching page of the Computer Engineering / Technische Informatik research group.

The labs include the implementation of learning strategies for run-time adaptable hardware signal classifiers, evolution of adaptable processor caches, and optimization of chip designs.

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

nächster Wiederholungstermin:

SS 2018

Homepage:

<https://cs.uni-paderborn.de/index.php?id=60192>

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 2018

weiterführende Veranstaltungen:

Projektgruppe

Homepage:

Unterlagen etc. siehe Paul

Empirical Performance Evaluation

Dozent: Karl

Büro: O3.158

Sprechstunde: Montag, 13 - 14 Uhr

Inhaltsangabe

The lecture “Empirical performance evaluation” deals with the needs and approaches of evaluating the performance of a system. We will consider approaches based on experiments and simulation. In the simulation part, we shall discuss typical approaches for discrete event simulation, suitable to a wide class of system from information and communication technology.

During the lecture, we will start out from a very simple simulation, and develop it step by step into a more general simulation framework.

This practical part of the lecture is supported by a part on the relevant statistical techniques: How to put randomness into a simulation, how to model assumptions about loads, faults, etc. Finally, techniques to evaluate results from simulation runs or experiments are discussed, along with structured approaches to plan larger simulation/experimental campaigns.

As a result of this class, a participant is able to design, develop, run, and interpret a performance evaluation task for non-trivial systems, on a level of expertise that gives statistically justified conclusions about the results.

The class is accompanied by exercises where both code development and statistical tools are tried out in practice.

Verschiedenes

Hörerkreis:

Master

nächster Wiederholungstermin:

SS 2018

Modulzugehörigkeit:

Modul III.3.1 und III.3.3

Homepage:

<https://cs.uni-paderborn.de/cn/teaching/courses/>

Algorithms for Synthesis and Optimization of Integrated Circuits

Dozent: Ghasemzadeh Mohammadi

Büro: O3. 134

Sprechstunde: Di 13:00 - 15:00

Inhaltsangabe

- Hardware modeling languages
- High-level synthesis and optimization methods (i.e., scheduling and binding)
- Logic representation and optimization of two-level logic functions
- Data structures for logic synthesis (Binary decision diagrams)
- Representation and optimization of multiple-level logic networks (Algebraic methods, controllability and observability computation, and timing verification)
- Modeling and optimization of sequential logic networks (Retiming)
- Libraries and binding

Literaturangaben

- **Giovanni De Micheli:** *Synthesis and optimization of digital circuits*, McGraw-Hill, Higher Education, 1994.

Verschiedenes

Hörerkreis:

Computer Science

Prüfungsform:

written examination, computer labs, homework assignments

vorausgesetzte Kenntnisse:

Principle logic design

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 2018

Modulzugehörigkeit:

Module ESS 3.1 und ESS 3.3

Homepage:

<https://cs.uni-paderborn.de/cn/teaching/courses/future-internet/>

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 be already an efficient programmer, this course is not a course to learn programming, but rather to use such skills.

Literaturangaben

- **M. Ward, G. Grinstein, D. Keim:** *Interactive Data Visualization*, 2010, AK Peters

Verschiedenes

Hörerkreis:

Master Informatik

Prüfungsform:

Submit assignments and oral exam

nächster Wiederholungstermin:

summer 2018

Homepage:

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

Modulzugehörigkeit:

Module II.4.1 “Computer Graphics and Visualization”

vorausgesetzte Kenntnisse:

Introduction to Computer Graphics

Vorbesprechung:

first lecture

Seminar: Algorithmic Complexity (in English)**Dozent:** Blömer**Büro:** F2.101**Sprechstunde:** Mi, 13 - 14 Uhr**Inhaltsangabe**

The seminar will cover selected topics from algorithm design and complexity. It will focus on two areas: First, algorithms and complexity theoretic results for clustering and facility location problems. Second, results for problems in lattice and coding theory. The first topic has numerous applications in combinatorial optimization, machine learning and data mining. The second topic is of interest for provable secure construction of cryptographic primitives that can even withstand attacks based in quantum algorithms. The seminar will be based on recent papers in the two above mentioned areas. The seminar is expected to take place as a block seminar at the end of the lecture period.

Literaturangaben

tba at the initial meeting

Verschiedenes**Hörerkreis:**

Informatik Master

Prüfungsform:

Presentation and Essay

vorausgesetzte Kenntnisse:

complexity theory; advanced algorithm design

Modulzugehörigkeit:

III.2.1, III.2.2, III.2.3, und III.2.4

Leistungspunkte:

4

nächster Wiederholungstermin:

SS2018

Master Seminar on Selected Topics in Wireless Netw

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 2017/2018

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 a seminars by students of the faculty of Arts and Humanities. This content is 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. Agile methods 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

nächster Wiederholungstermin:

WS2017/18

Modulzugehörigkeit:

SWT

Homepage:

<http://is.uni-paderborn.de/fachgebiete/fg-engels/lehre/ss15/hip-app/pg-hip-app.html>

PG Secure Integration of Cryptographic Software

Dozent: Bodden

Büro: ZM1.02-09

Sprechstunde: Nach Absprache

Inhaltsangabe

Cryptographic APIs are often being misused. Goal of this project group is to develop a tool that provides extensive tool support to application developers with only little or no experience in cryptography.

Verschiedenes

Hörerkreis:

Master Computer Science

Modulzugehörigkeit:

III.5.1 Projekt Group

Prüfungsform:

active cooperation, seminar talk, seminar thesis, design, implementation, presentation of the results

vorausgesetzte Kenntnisse:

Bachelor Computer Science, Java, Eclipse Plugin Development

weiterführende Veranstaltungen:

Master Thesis

Project Group: Trusted Communication Modules

Dozent: Scheideler

Büro: F2.326

Sprechstunde: Do, 16-17 Uhr

Inhaltsangabe

The aim of the project group is to develop a prototype of a trusted communication module. The purpose of a trusted communication module is to make distributed systems far more robust and secure than it was possible so far. It does that by providing a simple, protected communication layer that resides between the computer and its connection to the Internet. In a seminar phase, the members of the project group will familiarize themselves with the state of the art on protocols for robust communication systems and perform isolated experiments with them. In the implementation phase, the protocols identified by the various members will be integrated into the trusted communication module, and tests will be run with it to demonstrate its robustness.

Verschiedenes

Hörerkreis:

i-m

Prüfungsform:

SS 2017: Presentation and isolated experiments

Vorbesprechung:

see the website

Modulzugehörigkeit:

Info 3. Studienabschnitt, MUA

vorausgesetzte Kenntnisse:

Recommended: Advanced Distributed Algorithms and Data Structures

Homepage:

<http://cs.uni-paderborn.de/ti/lehre/veranstaltungen/ss-2017/tcm-project/>

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 Bachelorarbeiten und Masterarbeiten. 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:

- Spieltheorie
- Theorie paralleler Systeme
- Randomisierte Algorithmen
- Komplexitätstheorie
- Drahtlose Netzwerke
- Algorithmische Geometrie
- Algorithmen in der Computergrafik

Verschiedenes

Hörerkreis:

Bachelor, Master Informatik

vorausgesetzte Kenntnisse:

keine

Homepage:

www.hni.upb.de/alg/oberseminar

Prüfungsform:

kein Schein

nächster Wiederholungstermin:

Seminar findet jedes Semester statt

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>

Oberseminar: Theoretische Informatik

Dozent: Scheideler

Büro: F2.326

Sprechstunde: Do, 16-17 Uhr

Inhaltsangabe

Im Oberseminar werden neueste Forschungsergebnisse im Bereich der theoretischen Informatik vorgestellt. Sie richtet sich in erster Linie an interessierte Master Studenten und wissenschaftliche Mitarbeiter.

Verschiedenes

Hörerkreis:

i-m

vorausgesetzte Kenntnisse:

Fortgeschrittenes Wissen im Bereich der theoretischen Informatik.

nächster Wiederholungstermin:

WS 2017

Homepage:

<http://cs.uni-paderborn.de/ti/lehre/oberseminar/ss-2017/>

Fachdidaktische Grundlagen

Dozent: Schulte

Büro: F2.116

Sprechstunde: Mo, 13-14

Inhaltsangabe

1. Grundfragen der Informatikdidaktik
 - Aufgaben und Methoden der Informatikdidaktik
 - Informatikdidaktik und Fachwissenschaft Informatik
 - Fachwissenschaftliches und Fachdidaktisches Selbstverständnis der Informatik
 - Informatiksysteme aus didaktischer Perspektive
2. Informatikspezifische Medien und Methoden
 - Sprachkonzepte im Informatikunterricht
 - Medien im Informatikunterricht
 - Formen der Unterrichtsorganisation im Informatikunterricht
 - Technische Aspekte von Informatiksystemen im Informatikunterricht
3. Informatische Bildungskonzepte
 - Informatikunterricht und Allgemeinbildung
 - Informatikunterricht und Medienbildung
 - Informatikunterricht in der Sek I
 - GI-Gesamtkonzept informatischer Bildung / Bildungsstandards
4. Grundlagen der fachspezifischen Diagnostik im Informatikunterricht
 - Informatikunterricht und Lerntheorien
 - Leistungsdiagnostik im Informatikunterricht
 - Evaluation von Informatikunterricht

Verschiedenes

Hörerkreis:

Informatik Lehramt Bachelor (GyGe, BK, HRGe)

Prüfungsform:

mündliche Prüfung

weiterführende Veranstaltungen:

SUM

Vorbesprechung:

Erste Semesterwoche

Modulzugehörigkeit:

DGI Didaktische Grundlagen des Informatikunterrichts

Leistungspunkte:

3

nächster Wiederholungstermin:

Nächstes Sommersemester

Homepage:

<https://cs.uni-paderborn.de/ddi/lehre/veranstaltungen/sommersemester-2017>

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

Modulzugehörigkeit:

Modul 9: Einführung in die Informatik für Geisteswissenschaftler

Prüfungsform:

Klausur, Mini-Projekt, aktive Teilnahme an den Übungen

Leistungspunkte:

8

vorausgesetzte Kenntnisse:

keine

nützliche Parallelveranstaltungen:

keine

weiterführende Veranstaltungen:

Einführung in Web-bezogene Sprachen

nächster Wiederholungstermin:

Sommer 2018

Vorbesprechung:

Vorlesung in der ersten Semesterwoche

Homepage:

koala.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 und Dokumentsprachen zur Informationsmodellierung und -verarbeitung. 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 2018

Homepage:

Die URL wird in der Vorlesung bekannt gegeben.

Gesellschaft und Informationstechnik

Dozent: Selke

Büro: F1.101

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Veranstaltung gibt einen Überblick über die gesellschaftlichen Aspekte der Informationstechnik und versetzt die Studierenden in die Lage, die gesellschaftliche und ethische Bedeutung des Fachs zu beurteilen und Konsequenzen für verantwortungsbewusstes Handeln zu ziehen. Die Studierenden verstehen die Zusammenhänge zwischen Informationstechnik und Gesellschaft, sind in der Lage, die Auswirkungen informationstechnischer Produkte und Dienstleistungen zu analysieren und potentielle Konfliktfelder zu identifizieren, und kennen ethische Verhaltensregeln, wie sie zum Beispiel von den Berufsorganisationen IEEE und ACM herausgegeben werden.

Die Veranstaltung umfasst unter anderem folgende Inhalte: Geschichte der Informatik, Einführung in Ethik, Kommunikation in Netzwerken, Geistiges Eigentum, Datenschutz, Sicherheit von Computern und Netzwerken, Zuverlässigkeit von Computern, Professionelle Verantwortung, Wechselwirkungen zwischen Informatik und Gesellschaft.

Literaturangaben

- **Michael J. Quinn:** *Ethics for the Information Age*, 6th edition, Global Edition, Pearson Education, 2015

Verschiedenes

Hörerkreis:

Computer Engineering (BA)

Prüfungsform:

Klausur

nützliche Parallelveranstaltungen:

Rechtliche Grundlagen für IT-Berufe

Vorbesprechung:

Vorlesung in der ersten Semesterwoche

Modulzugehörigkeit:

Modul Recht und Gesellschaft

Leistungspunkte:

3

nächster Wiederholungstermin:

Sommer 2018

Homepage:

koala.uni-paderborn.de

5 Raum für Notizen

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					