

Universität Paderborn



Informatik
**Veranstaltungs-
Kommentar**

Für

Informatik ▷ Bachelor
▷ Master

Computer Engineering

Ingenieurinformatik

Lehramt Informatik GyGe

Für das WiSe 16/17

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	12
5	Raum für Notizen	65

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.: Jan Beinke

ISSN: 1868-0690

Redaktion: Dennis Baurichter, Jan Beinke, 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/>

Dennis Baurichter, Jan Beinke, Alex Wiens
V-Kom-Redaktion für das WiSe 2016

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

Karl	Grundlagen der Programmierung 1	(1.1)	12
Sauer	Softwareentwurf	(1.2)	13
Blömer / Hüllermeier	Modellierung	(2.1)	14
Meyer auf der Heide	Einführung in Berechenbarkeit, Komplexität und Formale Sprachen	(2.3)	16
Platzner	Grundlagen der Rechnerarchitektur	(3.1)	--

Hauptstudium Bachelorstudiengang

Pfahler	Programming Languages and Compilers	(1.1)	--
Bodden	Modellbasierte Softwareentwicklung	(1.1)	19
Feldmann	Fundamental Algorithms	(2.1)	21
Blömer	Einführung in Kryptographie	(2.1)	23
Karl	Verteilte Systeme	(3.1)	24
Karl	Rechnernetze	(3.1)	25
Dressler	Networked Embedded Systems	(3.1)	26
Dressler	History of Computer Systems	(3.1)	--
Domik	Grundlagen der Computergrafik	(4.1)	27
Szwillus	Usability Engineering	(4.1)	--

Proseminare

Domik	Interaktion mit medizinischen Volumendaten	29
Dressler	Distributed embedded Systems	--
Weiland	Praktische Umsetzung der barrierefreien Gestaltung	--
Scheideler	Effiziente Algorithmen	30

Masterstudiengang

Lettmann	Heuristic Search Algorithms	(1.4,2.1,2.2)	31
Bodden	Android Systems Security	(1.1,1.5)	32
Meyer	Model-Driven Software Development	(1.1,1.2,1.6)	34
Hüllermeier	Machine Learning II	(1.4,3.2)	35
Wehrheim	Model checking	(1.1,1.5)	36
Anjorin	Fundamentals of Model-Driven Engineering	(1.1,1.2,1.5,1.6)	37
Fischer	Algorithms for Highly Complex Virtual Scenes	(2.1,2.2)	39
Scheideler	Advanced Distributed Algorithms and Data Structures	(2.1,2.2,2.4)	40
Skopalik	Algorithmic Game Theory	(2.1,2.2,2.4)	41
Dressler	Operating Systems	(3.1,3.2,3.4)	42
Hamann	Evolutionary Robotics	(3.2,3.4,3.6)	43
Karl	Mobile communication	(3.1,3.3)	44
Platzner	Advanced Computer Architecture	(3.4,3.5)	--
Plessl	High-Performance Computing	(3.1,3.2,3.5)	45
Keil	Konzepte digitaler Medien	(4.2)	46
Tauber	Assistive Technologies, Accessibility	(4.2,4.3,4.5)	--
Selke	Einführung in Informatik und Gesellschaft	(4.2)	47
Szwilius	Usability Engineering Practice	(4.3,4.5)	--

Seminare

Wehrheim	Formal Models for Safe and Secure Systems (1.1,1.5,1.6)	48
Bodden	Secure Systems Engineering (1.1,1.2,1.5,1.6)	49
Scheideler	PRISMA (2.1,2.2)	--
Meyer auf der Heide	Gems of Theoretical Computer Science (2.1-2.4)	50
Blömer	Complexity and Cryptography (2.2,2.3)	51
Dressler	Selected Topics in Wireless Networking (3.1-3.4,3.6)	52
Selke	Geschichte interaktiver Systeme (4.1-4.7)	53

Projektgruppen

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

Dressler	Vehicular Networking	--
Bodden	Secure Integration of Cryptographic Software	54
Oberthür	History in Paderborn App 4	55
Busa-Fekete / Hüllermeier	Machine Learning	--
Dressler	Wireless Networking	56

Oberseminare

Hüllermeier	Maschinelles Lernen (SWT&IS)	--
Wehrheim	Spezifikation und Modellierung (SWT&IS)	--
Kleine Büning	Wissensbasierte Systeme (SWT&IS)	--
Bodden	Softwaretechnik (SWT&IS)	--
Engels	Informationssysteme (SWT&IS)	--
Scheideler	Theoretische Informatik (MuA)	57
Meyer auf der Heide	Algorithmen und Komplexität (MuA)	58
Blömer	Codes & Kryptographie (MuA)	--
Meyer auf der Heide	SFB 901 On-The-Fly Computing (MuA)	59
Domik	Interactive 3D Graphics and Games (MuA/MMWW)	--
Karl	Rechnernetze (ESS)	60
Platzner	Technische Informatik (ESS)	--
Dressler	Distributed Embedded Systems (ESS)	--

Platzner	Approximate Computing	(ESS)	--
Plessl	Hochleistungs-IT-Systeme	(ESS)	--
Keil	Informatik und Gesellschaft	(MMWW)	--
Domik	Computergrafik, Visualisierung und Bildverarbeitung	(MMWW)	--
Szwillus	Mensch-Computer-Interaktion	(MMWW)	--

Didaktik der Informatik

Schulte	Stufenbezogene Unterrichtsmodelle GymGe		--
Schulte	Stufenbezogene Unterrichtsmodelle BK		--
Schulte	Stufenbezogene Unterrichtsmodelle HRGe		--
Schulte	Methoden im Informatikunterricht		--
Schulte	Fachdidaktische Konzepte		--
Schulte	Informatiklernwerkstatt HRGe		--
Schulte	Informatiklernlabor GymGe und BK		--
Schulte	Begleitforschungsseminar Praxissemester		--
Schulte	Stufenbezogene Unterrichtsmodelle 2 BK		--
Schulte	Stufenbezogene Unterrichtsmodelle 2 GymGe		--
Schulte	US 1 (Lernwerkstatt) HRGe		--

Lehrveranstaltungen für andere Studiengänge

Fischer	Grundlagen der Programmierung für Ingenieure II		61
Fischer	Grundlagen der Programmierung für MB		62
Pfahler	Einführung in die Web-bezogenen Sprachen		63
Fischer	Datenverarbeitung		64

Allgemeine Veranstaltungen

n/a	Tutorenschulung Informatik		--
Plessl	PC ² Kolloquium: Kolloquium der Parallelverarbeitung		--

4.2 Informatik

Grundlagen der Programmierung 1

Dozent: Karl

Büro: O3.158

Sprechstunde: Mo, 13-14

Inhaltsangabe

Die Veranstaltung vermittelt grundlegende Programmier Techniken und -sprachen ebenso wie erste Techniken zur Erstellung größerer Programme. Dabei wird die Vorlesung zunächst mit der Sprache Python beginnen und dort einfache Techniken (Schleifen, Funktionen, etc.) besprechen und das Konzept der objektorientierten Programmierung einführen. Zum Ende der Vorlesung wird der dynamisch typisierten Sprache Python eine statisch typisierte Sprache - Java - gegenüber gestellt und die Vor- und Nachteile verglichen.

Lernziel ist zum Ende der Vorlesung in beiden Programmierstilen arbeitsfähig zu sein und einfache bis mittelgroße Programme darin entwerfen und implementieren zu können.

Verschiedenes

Hörerkreis:

Bachelor Informatik, Computer Engineering,
u.a.

Modulzugehörigkeit:

GP1

Prüfungsform:

Klausur

Leistungspunkte:

8

vorausgesetzte Kenntnisse:

Keine

nützliche Parallelveranstaltungen:

Modellierung

weiterführende Veranstaltungen:

Grundlagen der Programmierung 2, Grundlagen der Programmiersprachen

nächster Wiederholungstermin:

WS 17/18

Homepage:

<https://cs.uni-paderborn.de/index.php?id=59073>

Softwareentwurf

Dozent: Sauer

Büro: ZM1.03-44

Sprechstunde: nach Vereinbarung

Inhaltsangabe

In der Vorlesung Softwareentwurf wird der strukturierte Softwareentwurfsprozess unter Verwendung der Unified Modeling Language (UML) eingeführt. Das Lernziel der Veranstaltung ist es, das Vorwissen der Studierenden über die Programmierung „im Kleinen“ in ein methodisches Vorgehen einzubetten, das es ihnen erlaubt, größere Entwicklungsprojekte mit vorbereitenden Planungsaktivitäten und in Teams zu realisieren.

Im Einzelnen werden dabei in der Veranstaltung die Themen Anforderungsdefinition, Objektorientierte Analyse und Systementwurf vertiefend behandelt. Eingebettet in den Entwurfsprozess erwerben die Studierenden Kenntnisse der UML, vor allem Klassen-, Objekt-, Aktivitäten- und Sequenzdiagramme sowie Use Cases und Statecharts.

Literaturangaben

Vorlesungsunterlagen und weitere Materialien werden über das Web zur Verfügung gestellt.

Verschiedenes

Hörerkreis:

(Wirtschafts-)Informatiker, diverse andere Fächer mit Informatik-Anteil

Modulzugehörigkeit:

I.1.2 Softwaretechnik

Prüfungsform:

Klausur

vorausgesetzte Kenntnisse:

Grundlegende Kenntnisse in Java (z.B. durch „Grundlagen der Programmierung 1+2“)

weiterführende Veranstaltungen:

Softwaretechnikpraktikum (SWTPRA)

nächster Wiederholungstermin:

Wintersemester 2017/18

Homepage:

<http://is.uni-paderborn.de/fachgebiete/fg-engels/lehre/ws1617/softwareentwurf/aktuelles.html>

Modellierung

Dozent: Blömer

Büro: F2.101

Sprechstunde: Mi, 11-12 Uhr

Inhaltsangabe

Information zum Modul:

Das Modul Modellierung ist unter der Nummer I.2.1. im Modulhandbuch für den Bachelor Informatik detailliert in Bezug auf seine Rolle innerhalb des Studiengangs, auf die Inhalte und die Lernziele beschrieben. Es ist ein Pflichtmodul in den Bachelorstudiengängen Informatik und Wirtschaftsinformatik (1. Semester) und Pflichtveranstaltung im Studiengang Ingenieurinformatik und im Lehramtsstudium (3. Semester) und im Diplomstudiengang der Medienwissenschaften mit Schwerpunkt Medieninformatik.

Inhaltliche Gliederung der Veranstaltung

1. Einführung: Begriffe Modell, Modellierung
2. Grundlegende Formalismen: Mengen, Relationen, Funktionen, induktive Definitionen und Beweise
3. Logik: Aussagenlogik, Prädikatenlogik, Kalküle, Modellierung und Beweise
4. Modellierung mit Graphen: Weg, Verbindung, Zuordnung, Abhängigkeiten, Abfolgen, Fluss
5. Modellierung von Strukturen: reguläre Ausdrücke und kontextfreie Grammatiken
6. Modellierung von Abläufen: endliche Automaten

Umfang und Zeiten:

Der Arbeitsaufwand einschließlich der Prüfungsleistungen ist mit 10 Leistungspunkten festgelegt und entspricht durchschnittlich 300 Stunden individueller Arbeitszeit. Präsenzstunden sind 4 Vorlesungsstunden (V4) und 4 Übungsstunden (Ü4).

Dozenten:

Diese Veranstaltung wird von den Herren Blömer und Hüllermeier gemeinsam gehalten.

Literaturangaben

- **Kastens und Kleine Büning:** *Modellierung*, Hanser Verlag, 2. Auflage, 2008

Verschiedenes

Hörerkreis:

Bachelor (Ing.-) Informatik, Bachelor Wirtschaftsinformatik, Lehramt Informatik, Diplom Medienwissenschaften (Medieninformatik)

Modulzugehörigkeit:

BA Informatik: Gebiet I.2 Modelle und Algorithmen

Prüfungsform:

Klausur, Bekanntgabe zusätzlicher Modalitäten zu Beginn des Semesters

weiterführende Veranstaltungen:

Datenstrukturen und Algorithmen

Homepage:

L.079.05101

vorausgesetzte Kenntnisse:

Oberstufenstoff Mathematik, Bereitschaft und Fähigkeit zum Erlernen formaler Kalküle

nächster Wiederholungstermin:

WS 2017/18

Einführung in Berechenbarkeit, Komplexität und formale Sprachen

Dozent: Meyer auf der Heide

Büro: F1.301

Sprechstunde: Mi, 13-14 Uhr

Inhaltsangabe

Die Vorlesung wird eine Einführung in die Grundlagen der Berechenbarkeit, Komplexitätstheorie, und der formalen Sprachen geben. Konkrete Themen sind:

Berechenbarkeit

- Turingmaschinen
- entscheidbare und rekursiv aufzählbare Sprachen
- Churchsche These
- Unentscheidbarkeit
- Das Halteproblem
- Diagonalisierung
- Reduktionen

Komplexität

- Zeitkomplexität von Turingmaschinen
- Komplexitätsklassen
- Klassen P und NP
- NP-Vollständigkeit
- Das Erfüllbarkeitsproblem Boolescher Formeln
- Satz von Cook-Levin
- Polynomialzeit-Reduktionen
- Heuristiken
- Approximationsalgorithmen

Formale Sprachen

- reguläre Sprachen, reguläre Grammatiken
- endliche Automaten
- Minimierung endlicher Automaten
- reguläre Ausdrücke
- Pumping Lemma
- kontextfreie Sprachen, kontextfreie Grammatiken
- Kellerautomaten
- Chomsky Normalform
- CYK-Algorithmus
- kontextsensitive Sprachen, kontextsensitive Grammatiken
- linear beschränkte Automaten

Literaturangaben

- **Sipser:** *Introduction to the theory of computation*, PWS, ISBN: 0-619-21764-2, 978-0-619-21764-8
- **Garey, Johnson:** *Computers and intractability*, Freeman, ISBN: 0-7167-1044-7, 0-7167-1045-5
- **Hopcroft, Motwani, Ullman:** *Introduction to automata theory, languages, and computation*, Addison Wesley, ISBN: 978-0-321-47617-3, 0-321-47617-4
- **Wegener:** *Komplexitätstheorie*, Springer, ISBN: 3-540-00161-1
- **Wanka:** *Approximationsalgorithmen*, Teubner, ISBN: 3-519-00444-5, 978-3-519-00444-8

Verschiedenes

Hörerkreis:

i-b3, i-l5, ma3, s3

Prüfungsform:

Klausur

weiterführende Veranstaltungen:

- Komplexitätstheorie

Homepage:

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

Modulzugehörigkeit:

Info 1. Studienabschnitt, MUA

vorausgesetzte Kenntnisse:

keine

nächster Wiederholungstermin:

WS 2017/18

Programming Languages and Compilers

Dozent: Pfahler

Büro: F2.311

Sprechstunde: Nach Vereinbarung

Inhaltsangabe

Zu Zielen und Inhalt siehe Beschreibung des Moduls II.1.1 Softwaretechnik und Informationssysteme:

<http://www.cs.uni-paderborn.de/de/studierende/pruefungswesen/modulhandbuch.html>

Darin die Veranstaltung Programmiersprachen und Übersetzer sowie das elektronische Vorlesungsmaterial:

<http://ag-kastens.upb.de/lehre/material/plac2016>

Hinweis: Die Veranstaltung ist englischsprachig.

Literaturangaben

Siehe Vorlesungsmaterial:

<http://ag-kastens.upb.de/lehre/material/plac2016>

Verschiedenes

Hörerkreis:

i-b5

Prüfungsform:

Klausur

vorausgesetzte Kenntnisse:

Modul I.1.1 Programmiertechnik

Modulzugehörigkeit:

Modul II.1.1; SWT&IS

Leistungspunkte:

4

Homepage:

<http://ag-kastens.upb.de/lehre/material/plac2016>

Modellbasierte Softwareentwicklung

Dozent: Bodden

Büro: ZM1.02-09

Sprechstunde: Nach Absprache

Inhaltsangabe

Information zum Modul

Diese Veranstaltung ist eine mögliche Wahlpflichtveranstaltung im Modul Softwaretechnik und Informationssysteme im 2. Studienabschnitt des Bachelorstudiengangs Informatik. Sie kann darüberhinaus von Studierenden der Ingenieurinformatik und Wirtschaftsinformatik im Hauptstudium gehört werden.

Ziele

Die Studierenden sollen grundlegende Verfahren zur Konstruktion großer Softwaresysteme kennenlernen sowie gängige praxisrelevante Tools praktisch erproben (z.B. Together, UUPAL oder SPIN), die Vor- und Nachteile formaler und informaler Spezifikationstechniken erfahren und die Notwendigkeit von Design und abstrakter Repräsentation (Spezifikation) zur Verbesserung der Softwarequalität begreifen. Insbesondere wird auf das im Umfeld der UML postulierte Paradigma des "Model Driven-Development" (oder auch Model-Driven Architecture) eingegangen.

Teil I: Spezifikationstechniken für Analyse und Design

1. Strukturorientierte Techniken
 - Datenstrukturen: Design Pattern nach Gamma
 - Architekturen: Stile, Muster und
 - Beschreibungssprachen
2. Operationale Techniken
 - Statecharts: Syntax und Semantiken
 - Graphgrammatiken: Syntax und Semantik
3. Deskriptive Techniken: Z

Teil II Codegenerierung

4. Codegenerierung für Klassendiagramme,
 - Statecharts, Graphgrammatiken

Teil III Validation und Verifikation

5. Testen (Whitebox, Blackbox, Regressionsanalysen)
6. Der Einsatz und die Grundlagen von Model Checking

Literaturangaben

Folien sind im Netz verfügbar

Literatur:

- **Gamma et.al.:** *Design Patterns*, Addison-Wesley
- **Zündorf:** *Habilitation*, im Netz verfügbar
- **Ghezzi:** *Fundamentals of Software Engineering*, Addison Wesley
- **G. Berard et.al.:** *System and Software Verification*, Springer

Verschiedenes

Hörerkreis:

Bachelor Informatik, Ingenieurinformatik

Modulzugehörigkeit:

Softwaretechnik u. Informationssysteme

Prüfungsform:

je nach Teilnehmerzahl mündliche Prüfung
oder Klausur

vorausgesetzte Kenntnisse:

Vordiplom bzw. 1. Studienabschnitt Informatik oder Ingenieurinformatik, insbesondere SWE I,II GdP, TSE I,II, Modellierung

weiterführende Veranstaltungen:

zu jedem obigen Kapitel gibt es eine Reihe von weiterführenden Veranstaltungen in mehreren Modulen des Masterstudiengangs Informatik

nächster Wiederholungstermin:

WS 2017/18

Homepage:

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

Fundamental Algorithms

Dozent: Feldmann

Büro: F2.401

Sprechstunde: nach Vereinbarung

Inhaltsangabe

We will cover the following topics:

1. Advanced data structures
 - Binomial Heaps and Fibonacci Heaps
 - Advanced Search Structures
2. Graphalgorithms
 - Shortest Paths
 - Matchings
3. Network Flow
 - Ford-Fulkerson Algorithm
 - Preflow-Push Algorithm
4. Stringmatching
 - Knuth-Morris-Pratt Algorithm
 - Boyer-Moore Algorithm
 - Aho-Corasick Algorithm

Literaturangaben

- **T.H. Corman and C.E. Leiserson and R.L. Rivest:** *Introduction to Algorithms*, MIT Press, 1990
- **R.K. Ahuja and T.L. Magnanti and J.B. Orlin:** *Network flows*, Prentice Hall, 1993
- **G.A. Stephen:** *String searching algorithms*, World Scientific Publishing, 1994
- **S.O. Krumke, H. Noltemeier:** *Graphentheoretische Konzepte und Algorithmen*, Teubner, 2005
- **A.V. Aho and J.E. Hopcroft and J.D. Ullman:** *Data Structures and Algorithms*, Addison-Wesley, 1987
- **M.R. Garey and D.S. Johnson:** *Computers and intractability*, Freeman, 2000

Verschiedenes

Hörerkreis:

Bachelor, 2. Studienabschnitt

Modulzugehörigkeit:

Informatik Bachelor, 2. Studienabschnitt,
MUA

Prüfungsform:

written exam

vorausgesetzte Kenntnisse:

Knowledge about algorithms and data structures that is presented in the first four semesters of the Computer Science Bachelor studies in Paderborn.

Homepage:

in PAUL

Leistungspunkte:

4

weiterführende Veranstaltungen:

Methoden des Algorithmenentwurfs

Einführung in Kryptographie

Dozent: Blömer

Büro: F2.101

Sprechstunde: Mi, 11-12 Uhr

Inhaltsangabe

Die Kryptographie beschäftigt sich mit Methoden zur sicheren Datenübertragung. Moderne Kryptographie ist eine Schlüsseltechnologie mit vielen Anwendungen, von der EC-Karte, Mobiltelefon, TV-Decodern und elektronischem Geld bis zur fälschungssicheren elektronischen Unterschrift auf Bestellungen und Verträgen im Internet.

In dieser Vorlesung werden einige der grundlegenden modernen Kryptosysteme wie AES und RSA vorgestellt. Weiter werden die wichtigsten Sicherheitskonzepte der modernen Kryptographie diskutiert.

Literaturangaben

- **Buchmann:** *Einführung in die Kryptographie*, 5. Auflage, Springer, 2010. ISBN: 978-3-642-11185-3
- **Katz, Lindell:** *Introduction to Modern Cryptography*, Chapman & Hall / CRC Press, 2007. ISBN: 1-5848-8551-3
- **Stinson:** *Cryptography: Theory and Practice*, 2nd edition, Chapman & Hall / CRC Press, 2001. ISBN: 1-5848-8206-9

Verschiedenes

Hörerkreis:

Bachelor Informatik

Prüfungsform:

je nach Teilnehmerzahl Klausur oder mündliche Prüfung

weiterführende Veranstaltungen:

Cryptography: Provable Security
Cryptographic Protocols

Homepage:

L.079.05601

Modulzugehörigkeit:

MUA

vorausgesetzte Kenntnisse:

Lineare Algebra, nützlich sind auch Kenntnisse in Algebra und Zahlentheorie

nächster Wiederholungstermin:

voraussichtlich in einem Jahr

Verteilte Systeme

Dozent: Karl

Büro: O3.158

Sprechstunde: Montag, 13 - 14 Uhr

Inhaltsangabe

Diese Veranstaltung gehört zum Modul II.3.1 Eingebettete Systeme und Systemsoftware (Pflichtmodul im Gebiet „Eingebettete Systeme und Systemsoftware“). In diesem Modul wird eine breite Einführung in die Gebiete Eingebettete Systeme, HW/SW Codesign, Verteilte Systeme und in Rechnernetze vermittelt.

Inhaltliche Informationen zur Veranstaltung „Grundlagen der Verteilten Systeme“
Diese Vorlesung vermittelt Kenntnisse über die Architektur und Funktionalität von Verteilten Systemen, die eine wichtige Komponente komplexer Anwendungssysteme bilden. Dabei werden charakteristische Eigenschaften und Systemmodelle sowie unterstützende Aspekte aus den Bereichen Rechnerkommunikation, Betriebssysteme und Sicherheit betrachtet. Nach der Vorstellung der klassischen und erweiterten Client/Server-Elementen, Sockets und Request/Reply-Protokollen werden entfernte Objektaufrufe behandelt und an konkreten Beispielen von JavaRMI, Corba und .NET verdeutlicht. Die Vorlesung schließt mit der Betrachtung von Namens- und Erkennungsdiensten.

Verschiedenes

Hörerkreis:

Bachelor

Prüfungsform:

Klausur

nächster Wiederholungstermin:

SS 2017

Modulzugehörigkeit:

ESS

vorausgesetzte Kenntnisse:

Konzepte und Methoden der Systemsoftware

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/fachgebiet-rechnernetze/lehre/ws1112/verteilte-systeme.html>

Rechnernetze

Dozent: Karl

Büro: O3.158

Sprechstunde: Montag, 13 - 14 Uhr

Inhaltsangabe

Die Veranstaltung beschreibt die Grundlagen, Architekturen und Protokolle der Rechnernetze und Telekommunikationsnetze, mit besonderem Schwerpunkt auf Internet-basierten Ansätzen. Die Veranstaltung vertieft die Vorkenntnisse aus KMS. Wesentliche Themen der Veranstaltung sind:

- Grundlegende Abstraktionen - Protokoll, Service, Layering, Multiplexing, Scheduling, Duplex, ...
- Architekturmodelle: ISO/OSI, Internet
- Programmierschnittstelle
- Grundlagen der Nachrichtentheorie, Eigenschaften der physikalischen Übertragung
- Vielfachmedienzugriff (Medium Access Control) und Sicherungsschicht
- Wegewahl, Routing, Forwarding; Netzstrukturen
- Überlastabwehr, Flußkontrolle.

Mehr über die Inhalte erfahren Sie auch in den Foliensätzen der ersten beiden Kapitel.

Literaturangaben

- **A. Tanenbaum:** *Computer Networks*, 4. Auflage, Prentice Hall

Alternativ

- **L. L. Peterson & B. S. Davie:** *Computer Networks: Systems Approach*, 2003, 3rd edition, Morgan Kaufman
- **J. F. Kurose & K. W. Ross:** *Computer Networking: A Top-Down Approach Featuring the Internet*, 2004, 3rd edition, Addison Wesley

Verschiedenes

Hörerkreis:

Informatik Bachelor

Prüfungsform:

bestehen der Klausur

nächster Wiederholungstermin:

WS 2017/2018

Modulzugehörigkeit:

ESS

vorausgesetzte Kenntnisse:

KMS oder vergleichbar

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/fachgebiet-rechnernetze/lehre/lehre-ws1112/vl-rechnernetze.html>

Networked Embedded Systems (NES)

Dozent: Dressler

Büro: F1.401

Sprechstunde: nach Absprache

Inhaltsangabe

Ziel des Kurses ist es, ein tiefgehendes Verständnis in den Aufbau und die Programmierung vernetzter eingebetteter Systeme im Allgemeinen und in Sensornetzen im Speziellen zu vermitteln. Dazu werden in der Vorlesung entsprechende Grundlagen von Sensornetzen diskutiert, die dann im Rahmen der Übung praktisch umgesetzt werden sollen.

- Aufbau eines Sensorknotens
Architektur eingebetteter Systeme, Programmierparadigmen
- Sensornetze
Verknüpfung mehrerer Knoten zu einem Netz
- Routing
Einfache Routingalgorithmen
- Lokalisierung
Lokalisierungsmethoden, Signalstärke, Laufzeitmessungen

Literaturangaben

- **Falko Dressler:** *Self-Organization in Sensor and Actor Networks*, Chichester, John Wiley & Sons, 2007
- **Holger Karl and Andreas Willig:** *Protocols and Architectures for Wireless Sensor Networks*, Chichester, John Wiley & Sons, 2005

Verschiedenes

Hörerkreis:

Bachelor Informatik, Computer Engineering

vorausgesetzte Kenntnisse:

Rechnernetze

nächster Wiederholungstermin:

WS 2017/2018

Homepage:

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

Grundlagen Computergrafik

Dozent: Domik

Büro: F2.204

Sprechstunde: Bitte Webseite beachten.

Inhaltsangabe

Dies sind die Hauptthemen:

- Objekt Modellierung in der Grafik
- Transformationen (2d und 3d)
- Geometrische Projektionen
- Entfernen verdeckter Oberflächen (Hidden Surface Removal)
- Licht und Reflektion
- Schattierung
- Filling & Clipping
- Rasterung, Texturen

Programmiert wird mit OpenGL und Shader Sprachen unter Java.

Sie beginnen mit Punkten, Kanten und Polygonen und enden mit interaktiven Echtzeit-Grafiken.

Die erlernten Kompetenzen sind u.a.:

- Programmierung der Rendering Pipeline
- Praktische Anwendung der Linearen Algebra
- Einbettung von 3D Modellen in Ihre Programme
- Erzeugung interaktiver Echtzeit-Grafik

Weiteres:

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

Literaturangaben

- **Ed Angel:** *Interactive Computer Graphics*,

Verschiedenes

Hörerkreis:

Informatik Bachelor

Modulzugehörigkeit:

Info 2. Studienabschnitt, MMWW

Prüfungsform:

Übungen und Klausur

weiterführende Veranstaltungen:

(Pro)Seminare (B/M)

Data and Information Visualization (Master)

Vorbesprechung:

1. Vorlesungsstunde

vorausgesetzte Kenntnisse:

Linear Algebra.

Rechnen mit Vektoren (z.B. skalare Multiplikation, Vektorprodukt) und Matrizen

nächster Wiederholungstermin:

WS 17/18

Homepage:

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

Proseminar Interaktion mit medizinischen Volumendaten

Dozent: Domik

Büro: F2.204

Sprechstunde: Bitte Webseite beachten.

Inhaltsangabe

- HCI für medizinische Visualisierung
- Direct Volume Visualization
- Transferfunktionen in der medizinischen Visualisierung
- HCI für Volumendaten

Literaturangaben

- **B. Preim, Charl Botha:** *Visual Computing for Medicine*, Morgan Kaufmann/Elsevier, 2014

Verschiedenes

Hörerkreis:

Informatik Bachelor

Prüfungsform:

Mitarbeit, Vortrag, schriftliche Ausarbeitung

nützliche Parallelveranstaltungen:

Grundlagen Computergrafik

nächster Wiederholungstermin:

wahrscheinlich wieder WS

Homepage:

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

Modulzugehörigkeit:

Info, 2. Studienabschnitt, MMWW

vorausgesetzte Kenntnisse:

Informatik - ersten beiden Semester.

weiterführende Veranstaltungen:

Rendering; Visualization; Seminare in Grafik/Visualisierung

Vorbereitung:

Erstes Treffen - achten Sie auf meine Lehre-Webseite

Proseminar: Effiziente Algorithmen

Dozent: Scheideler

Büro: F2.326

Sprechstunde: Do, 16-17 Uhr

Inhaltsangabe

Es gibt bereits seit vielen Jahren einen renommierten internationalen Programmierwettbewerb für Studenten, den International Collegiate Programming Contest (ICPC). Bisher hat sich Paderborn dort leider rausgehalten. Das soll sich nun ändern. Ziel des Proseminars ist es, die Studierenden mit den Problemstellungen im ICPC vertraut zu machen und ihnen beizubringen, wie effiziente Verfahren für diese Problemstellungen entwickelt werden können. Dazu wird es wöchentliche Treffen an einem Rechnerpool geben, in denen in Gruppenarbeit Verfahren entwickelt, implementiert und eingereicht werden und anschließend besprochen werden. Den besten Teams wird bei Interesse die Möglichkeit gegeben, kostenfrei an einem ICPC-Wettbewerb teilzunehmen.

Die Proseminarnote wird sich daran orientieren, ob an den Treffen regelmäßig und aktiv teilgenommen wird und wie gut die Probleme (nach einer Einarbeitungszeit) gelöst werden können.

Literaturangaben

- **Steven Skiena and Miguel Revilla:** *Programming Challenges - The Programming Contest Training Manual*, Springer Verlag 2003

Verschiedenes

Hörerkreis:

i-b5, winf

Prüfungsform:

Erfolgreiche Teilnahme

nächster Wiederholungstermin:

WS 2017

Homepage:

<http://cs.uni-paderborn.de/ti/ws-2016/prosea.html>

Modulzugehörigkeit:

MUA

vorausgesetzte Kenntnisse:

Datenstrukturen und Algorithmen

Hilfreich: Grundlegende Algorithmen

Vorbesprechung:

Wird noch bekannt gegeben

Heuristic Search

Dozent: Lettmann

Büro: O4.173

Sprechstunde: nach Vereinbarung

Inhaltsangabe

This course is given in English.

Heuristic search algorithms are basis of solutions to many problems. Using domain knowledge can increase efficiency dramatically. This course gives an introduction into modeling search problems and presents GBF-algorithms (general-best-first) for AND/OR-graph search resp. BF-algorithms (best-first) for OR-graph search. Theoretical properties like completeness, (ϵ -)admissibility are proven and efficiency is analyzed with respect to heuristic functions in use.

Selected topics:

1. state space representation vs. problem reduction representation,
2. basic search algorithms,
3. informed best-first search,
4. specializations of best-first search,
5. formal properties of algorithms and heuristic functions,
6. relaxing optimality,
7. designing heuristics,
8. basic game playing strategies.

In the future, this course will be replaced by a new course on “Planning and Heuristic Search”.

Literaturangaben

- **J. Pearl:** *Heuristics*, Addison Wesley, 1983
- **St. J. Russel, P. Norvig:** *Artificial Intelligence: A Modern Approach*, Prentice Hall, 1995
- **N. J. Nilsson:** *Principles of Artificial Intelligence*, Springer, 1982
- **S. Edelkamp, S. Schroedl:** *Heuristic Search: Theory and Applications*, Elsevier, 2011

Verschiedenes

Hörerkreis:

(Ing.-)Informatik Master, Winfo Master

Modulzugehörigkeit:

Modul III.1.4 (SWT&IS Wissensbasierte Systeme, Kleine Büning) und III 2.1.(MuA Algorithmen, Meyer auf der Heide)

Prüfungsform:

oral exam

Leistungspunkte:

4 ECTS

Homepage:

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

Android System Security

Dozent: Bodden

Büro: ZM1.02-09

Sprechstunde: Nach Absprache

Inhaltsangabe

Android is the fastest growing mobile system in the world. With thousands of new applications submitted to the Google Playstore per day, many of them are written by novice programmers. These applications may contain serious security issues, which exposes millions of users to security threats. Apart from benign applications, malicious applications aim to steal private information or send expensive premium SMS messages to numbers owned by the malware authors. This course shows which security issues exist in applications in the wild and how to avoid them. Furthermore, we will teach techniques that malware uses to evade detection by both humans and malware-analysis tools. The analysis tools we discussed are based on recent research results. This course consists of a theoretical lecture and a practical part, in which hands-on experience can be gained. Theoretical and practical parts are evenly distributed throughout the course. For the practical part, exercises will be provided and can usually be solved in groups to gain bonus points.

Literaturangaben

- **Patrick Lam, Eric Bodden, Ondrej Lhoták and Laurie Hendren:** *The Soot framework for Java program analysis: a retrospective*, CETUS '11
- **Steven Arzt, Alexandre Bartel, Damien Ochteau et. al.:** *FlowDroid: Precise Context, Flow, Field, Object-sensitive and Lifecycle-aware Taint Analysis for Android Apps*, PLDI '14
- **Siegfried Rasthofer, Steven Arzt, Marc Miltenberger, Eric Bodden:** *Harvesting Runtime Values in Android Applications That Feature Anti-Analysis Techniques*, NDSS '16
- **W. Enck, P. Gilbert, B. gon Chun, L. P. Cox, J. Jung, P. McDaniel, and A. Sheth:** *Taintdroid: An information-flow tracking system for realtime privacy monitoring on smartphones*, OSDI 2010
- **K. Coogan, S. Debray, T. Kaochar, and G. Townsend:** *Automatic Static Unpacking of Malware Binaries*, WCRE 2009
- **William Enck, Damien Ochteau, Patrick McDaniel, and Swarat Chaudhuri:** *A Study of Android Application Security*, SEC 2011

Verschiedenes

Hörerkreis:
Informatik Master

Modulzugehörigkeit:
III.1 .5 Analytische Methoden des Software Engineering

Prüfungsform:
Oral exam/mündliche Prüfung

Leistungspunkte:
4

vorausgesetzte Kenntnisse:

A mature understanding of the Java programming languages and object-oriented programming will be helpful.

nächster Wiederholungstermin:

WiSe 2017/18

nützliche Parallelveranstaltungen:

The courses Software Analysis and/or Designing Code Analyses for Large-Scale Software Systems (DECA) are a recommended but not required prerequisite.

Homepage:

<https://www.hni.uni-paderborn.de/swt/lehre/android-system-security/>

Model-Driven Software Development (in English)

Dozent: Meyer

Büro: ZM1.02-10

Sprechstunde: auf Anfrage

Inhaltsangabe

Modellgetriebene Software-Entwicklung verfolgt die Entwicklung von Software-Systemen auf Basis von Modellen. Dabei werden die Modelle nicht nur für Dokumentation, Entwurf und Analyse eines initialen Systems verwendet, sondern dienen vielmehr als primäre Entwicklungsartefakte, aus denen das finale System nach Möglichkeit vollständig generiert werden kann.

Diese Zentrierung auf Modelle bietet eine Reihe von Vorteilen, wie z.B. eine Anhebung der Abstraktionsebene, auf der das System spezifiziert wird, verbesserte Kommunikationsmöglichkeiten, die durch domänenspezifische Sprachen (DSL) bis zum Endkunden reichen können, und eine Steigerung der Effizienz der Software-Erstellung durch automatisierte Transformationen der erstellten Modelle bis hin zum Quellcode des Systems. Allerdings gibt es auch noch einige, zum Teil ungelöste Herausforderungen beim Einsatz von modellgetriebener Software-Entwicklung wie beispielsweise Modellversionierung, Evolution von DSLs, Wartung von Transformationen oder verteiltes, kooperatives Arbeiten und MDSD. Obwohl aufgrund der genannten Vorteile MDSD in der Praxis bereits im Einsatz ist, bieten die genannten Herausforderungen auch noch Forschungsbedarf.

Die Vorlesung wird zu MDSD gehörende Konzepte und Techniken einführen. Als Grundlage wird dazu die systematische Erstellung von Meta-Modellen und DSLs einschließlich aller nötigen Bestandteile (konkrete und abstrakte Syntax, statische und dynamische Semantik) eingeführt. Anschließend erfolgt eine allgemeine Diskussion der Konzepte von Transformationsprachen sowie eine Einführung in ausgewählte Transformationssprachen. Die verbleibenden Vorlesungen behandeln weiterführende Fragestellungen, z. B. der Modellversionierung, Modellkopplung, MDSD-Standards, Teamarbeit auf Basis von Modellen, Testen von modellgetriebener erstellter Software, sowie der Wartung und Weiterentwicklung von Modellen, Meta-Modellen und Transformationen. Abschließend werden modellgetriebene Verfahren für die Erstellung und Analyse von Software für Intelligente Technische Systeme als weiterführende Einheit behandelt.

Die Vorlesung findet in englischer Sprache statt.

Literaturangaben

- **Tom Stahl, Markus Völter:** *Model-Driven Software Development*, John Wiley & Sons, 2006.

Verschiedenes

Hörerkreis:

Master Informatik

Prüfungsform:

Mündliche Modulprüfung

nützliche Parallelveranstaltungen:

Web Engineering

Modulzugehörigkeit:

Info Master, SWT

vorausgesetzte Kenntnisse:

Model-Based Software Engineering, Englisch

Homepage:

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

Machine Learning II (in English)

Dozent: Hüllermeier

Büro: O4.258

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Machine learning is “the art and science of algorithms that make sense of data” (Flach, 2012). 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 information systems.

This lecture will cover selected topics in contemporary machine learning research, such as reinforcement learning, online learning and bandit algorithms, multi-task learning, multi-target and structured output prediction, preference learning, learning from weak supervision, and uncertainty in machine learning. The focus of the lecture will be on methods and algorithms, though theoretical issues and applications will be addressed, too.

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:

(Ing.-)Informatik Master, Winfo Master

Modulzugehörigkeit:

Modul III.1.4 (SWT&IS Wissensbasierte Systeme, Kleine Büning)

Prüfungsform:

Mündliche Prüfung

Leistungspunkte:

4

vorausgesetzte Kenntnisse:

This lecture is a continuation of the “Machine Learning I” course. It assumes elementary knowledge of learning from data and model induction (theory of generalization), including methods for basic (supervised) learning tasks such as classification and

nächster Wiederholungstermin:

Wintersemester 2017/2018

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/intelligente-systeme>

Model checking

Dozent: Wehrheim

Büro: O4.225

Sprechstunde: nach Vereinbarung

Inhaltsangabe

In the course we will study techniques for automatically verifying that a system (software or hardware) is correct, i.e. adheres to requirements describing the desired functionality. For describing requirements a particular class of logics, so called temporal logics, will be employed. Temporal logics can be used to describe properties of systems in time. For this class of logics there are algorithms for checking whether a property does or does not hold for a system. If the system under consideration has a finite state space, tools implementing these algorithms can fully automatically carry out the verification.

In the course we will take a look at two temporal logics (LTL and CTL) and their model checking algorithms. We will furthermore work with model checking tools (in particular SPIN) and verify small examples of systems (for instance distributed algorithms) in the exercises.

The course will be taught in English.

Literaturangaben

- **E. Clarke et al.:** *Model checking*, MIT Press, 1999

Verschiedenes

Hörerkreis:

Master

Prüfungsform:

Mündliche Prüfung

nützliche Parallelveranstaltungen:

Deductive Verification
Software Analysis

Modulzugehörigkeit:

SWT & IS, Module 1.1 und 1.5

vorausgesetzte Kenntnisse:

Logik, Automatentheorie

nächster Wiederholungstermin:

noch nicht bekannt

Fundamentals of Model-Driven Engineering (in English)

Dozent: Anjorin

Büro: ZM1.03-06

Sprechstunde: Mo, 10-11 Uhr

Inhaltsangabe

Model-Driven Engineering (MDE) focusses on the use of models, i.e., suitable abstractions, as primary artefacts in (software) engineering processes. Important tasks include specifying these models (metamodelling), deriving new models from other models (model transformation), and keeping different but related models consistent to each other (model synchronisation).

While it is important to develop an intuitive understanding for central concepts such as models, metamodels, and model transformations, it is equally important to establish solid formal foundations for at least the basic concepts. This is especially the case when providing reliable tool support (e.g., static analyses) for MDE activities.

This lecture, therefore, introduces basic MDE concepts including models, metamodels, and model transformations, providing a precise and detailed formalisation using very basic category theory (algebraic graph transformation).

The lecture is designed to be especially accessible to computer scientists, by providing a hands-on constructive mapping of all definitions and results to executable programs in a main stream OO language (Java). The model-driven development of a simple project management tool (<https://trello.com>) will be used as a “mini-project” in the exercises to demonstrate and explore all concepts introduced in the lecture.

This lecture complements other MDSD lectures and can be taken before (providing foundations) or after (providing a specialization).

Literaturangaben

- Ehrig, H., Ehrig, K., Prange, U., Taentzer, G.: *Fundamentals of Algebraic Graph Transformation*, <http://www.springer.com/us/book/9783540311874>
- D.E. Rydeheard, R.M. Burstall: *Computational Category Theory*, <http://www.cs.man.ac.uk/~david/categories/book/book.ps>
- Ehrig, H., Ermel, C., Golas, U., Hermann, F.: *Graph and Model Transformation*, <http://www.springer.com/de/book/9783662479797>

Verschiedenes

Hörerkreis:

Informatik Master, Mathematik Master

Modulzugehörigkeit:

Informatik Master, SWT (III.1.1, III.1.2, III.1.5, III.1.6)

Prüfungsform:

Mündliche Prüfung

vorausgesetzte Kenntnisse:

Lineare Algebra (Grundkenntnisse), gute Kenntnisse einer modernen objektorientierten Programmiersprache (zB. Java).

nützliche Parallelveranstaltungen:

Model-Driven Software Development (in English)

weiterführende Veranstaltungen:

Model-Driven Software Development (in English)

Homepage:

[http://is.uni-paderborn.de/
fachgebiete/fg-engels/lehre/ws16/
fundamentals-of-model-driven-engineering](http://is.uni-paderborn.de/fachgebiete/fg-engels/lehre/ws16/fundamentals-of-model-driven-engineering)

Algorithms for highly complex virtual scenes

Dozent: Fischer

Büro: F1.223

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Walkthrough systems allow viewing and a walk through virtual 3D scenes. Such systems are used in architectural applications, simulations or rendering of CAD data. The efficiency of real-time rendering algorithms is crucial for a smooth and fast rendering of virtual 3D scenes in a walkthrough system. There are various approaches to simplify highly complex 3D geometric data and to achieve a rendering of the data in real time. In the lecture, we will learn algorithmic approaches in the areas of visibility culling, simplification, level of detail, point based rendering, image-based and parallel rendering.

Literaturangaben

- **Tomas Akenine-Möller, Eric Haines:** *Real-Time Rendering*, AK Peters, 2002
- **David Luebke, Martin Reddy, Jonathan D. Cohen:** *Level of Detail for 3D Graphics*, Morgan Kaufmann Publishers, 2002
- **Thomas Rauber:** *Algorithmen in der Computergraphik*, Teubner, 1993

Verschiedenes

Hörerkreis:

Master

Prüfungsform:

oral examination

weiterführende Veranstaltungen:

Master Thesis

Modulzugehörigkeit:

MuA Modul III 2.1, 2.2

vorausgesetzte Kenntnisse:

Basic data structures and algorithms of Bachelor studies, Fundamentals of computer graphics (Gitta Domik) are advantageous but not mandatory.

Homepage:

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

Fortgeschrittene verteilte Algorithmen und Datenstrukturen

Dozent: Scheideler

Büro: F2.326

Sprechstunde: Do, 16-17 Uhr

Inhaltsangabe

This Master-level course will be given in English and will cover advanced topics on distributed algorithms and data structures. More details will be given on the course webpage.

Literaturangaben

The lecture will be based on recent conference and journal publications.

Verschiedenes

Hörerkreis:

i-m

Prüfungsform:

Mündliche Prüfung

nächster Wiederholungstermin:

WS 2017

Modulzugehörigkeit:

Info 3. Studienabschnitt, MUA

vorausgesetzte Kenntnisse:

Bachelor-Vorlesung „Verteilte Algorithmen und Datenstrukturen“ empfohlen

Homepage:

<http://cs.uni-paderborn.de/ti/ws-2016/avads.html>

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:

mündl. Prüfung

Modulzugehörigkeit:

MuA III.2.1, III.2.2, III.2.4

Homepage:

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

Operating Systems

Dozent: Dressler

Büro: F1.401

Sprechstunde: nach Absprache

Inhaltsangabe

In this class, we discuss general aspects of operating systems as well as specific characteristics of real-time operating systems and operating systems for embedded systems.

Topics of interest include:

- Parallelism
- Scheduling
- Synchronization
- Inter-Process Communication
- Memory Management
- Security
- Embedded OS
- Real-Time

Literaturangaben

- **Andrew S. Tanenbaum:** *Modern Operating Systems*, ed. 3rd, Upper Saddle River, NJ, Pearson-Prentice Hall, 2009
- **William Stallings:** *Operating Systems: Internals and Design Principles*, ed. 8th, Upper Saddle River, Pearson Prentice Hall, 2014
- **Abraham Silberschatz, Peter B. Galvin and Greg Gagne:** *Operating Systems Concepts*, ed. 9th, John Wiley & Sons, 2013

Verschiedenes

Hörerkreis:

Master Informatik, Master Computer Engineering

nächster Wiederholungstermin:

Winter 2017/2018

Homepage:

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

Evolutionary Robotics (in English)

Dozent: Hamann

Büro: F1.221

Sprechstunde: (nach Vereinbarung)

Inhaltsangabe

Evolutionary robotics belongs to the field of artificial intelligence, in particular machine learning. “Evolutionary robotics is a new technique for automatic creation of autonomous robots. It is inspired by the Darwinian principle of selective reproduction of the fittest. It is a new approach which looks at robots as autonomous artificial organisms that develop their own skills in close interaction with the environment without human intervention. Heavily drawing from natural sciences like biology and ethology, evolutionary robotics makes use of tools like neural networks, genetic algorithms, dynamic systems, and biomorphic engineering.” (Nolfi and Floreano, “Evolutionary Robotics”)

“The basic idea behind evolutionary robotics goes as follows. An initial population of different artificial chromosomes, each encoding the control system (and sometimes the morphology) of a robot, are randomly created and put in the environment. Each robot (physical or simulated) is then let free to act (move, look around, manipulate) according to a genetically specified controller while its performance on various tasks is automatically evaluated. The fittest robots are allowed to reproduce (sexually or asexually) by generating copies of their genotypes with the addition of changes introduced by some genetic operators (e.g., mutations, crossover, duplication). This process is repeated for a number of generations until an individual is born which satisfies the performance criterion (fitness function) set by the experimenter.” (Nolfi and Floreano, “Evolutionary Robotics”)

In difference to the application of evolutionary algorithms in optimization, challenges in evolutionary robotics are autonomy, time efficiency, and the complexity of fitness space and search space.

This course covers the biological fundamentals, an overview of evolutionary algorithms, artificial neural networks, how to evolve robots, co-evolution, and a number of sophisticated techniques to improve the performance of the evolutionary robotics approach.

Literaturangaben

- **Stefano Nolfi and Dario Floreano:** *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*, MIT Press, 2000

Verschiedenes

Hörerkreis:

Master, Computer Science

Prüfungsform:

oral exam

nächster Wiederholungstermin:

WS 2017/18

Modulzugehörigkeit:

Eingebettete Systeme

Leistungspunkte:

4 ECTS

Homepage:

<http://upb.de/cs/si>

Mobilkommunikation

Dozent: Karl

Büro: O3.158

Sprechstunde: Montag, 13 - 14 Uhr

Inhaltsangabe

Die Vorlesung gibt einen Überblick über Systeme, Architekturen und Protokolle zur drahtlosen und mobilen Kommunikation. Behandelt werden unter anderem:

- Drahtlose Kommunikation, drahtlose Kanäle
- Medienzugriff in drahtlosen Medien
- Drahtlose lokale Netze (WLAN, IEEE 802.11)
- Systeme der zellularen Mobilkommunikation (GSM, UMTS)

Literaturangaben

- **J. Schiller:** *Mobilkommunikation (dpunkt) bzw. Mobile Communications (Addison Wesley)*, derzeit 2. Auflage, 3. Auflage bald erhältlich

Verschiedenes

Hörerkreis:

Master

Prüfungsform:

bestehen der Klausur

nächster Wiederholungstermin:

WS 2017/2018

Modulzugehörigkeit:

ESS 3.1 und ESS 3.3

vorausgesetzte Kenntnisse:

Rechnernetze

Homepage:

<http://www.cs.uni-paderborn.de/fachgebiete/fachgebiet-rechnernetze/lehre/lehre-ws-1112/mobkom1112.html>

High-Performance Computing

Dozent: Plessl

Büro: O2.167

Sprechstunde: By arrangement

Inhaltsangabe

- Einführung in High-Performance Computing
- Modelle und Programmiermuster für paralleles Rechnen
- Programmiersprachen und Bibliotheken für HPC
- Performanceanalyse, Optimierung und Debugging
- Heterogenes Rechnen mit Hardwarebeschleunigern
- Fallstudien

Verschiedenes

Hörerkreis:

Master Informatik, Master Computer Engineering

Modulzugehörigkeit:

Siehe PAUL

Prüfungsform:

mündliche Prüfung

Leistungspunkte:

4

vorausgesetzte Kenntnisse:

- Programmierkenntnisse in C/C++
- Empfohlen: Rechnerarchitektur

nützliche Parallelveranstaltungen:

Advanced Computer Architecture

nächster Wiederholungstermin:

WS 2017/18

Homepage:

<http://homepages.uni-paderborn.de/plessl/teaching/2016-HPC>

Konzepte digitaler Medien

Dozent: Keil

Büro: F1.428

Sprechstunde: n. V.

Inhaltsangabe

Die klassischen Medientheorien sind vorrangig Rezeptionsanalysen von Massenmedien unter der besonderen Berücksichtigung von Film und Fernsehen. Mathematische Formeln, technische Zeichnungen oder Verwaltungsformulare werden in der Medientheorie nicht betrachtet. Durch den Computer werden jedoch diese Grenzen aufgelöst. Digitale Medien verbinden potenziell alle bislang gekannten Medienformen, wenn auch nicht in der gleichen Qualität und mit den gleichen Produktions- und Rezeptionsbedingungen.

Über den Begriff des Zeichens und seiner Verarbeitung mit Hilfe von (digitalen) Automaten erhält man einen erweiterten Medienbegriff, der es gestattet, die Vielfalt digitaler Medien unter einem gemeinsamen technischen Bezugspunkt zu thematisieren. Analog zu den Konzepten von Programmiersprachen lassen sich unterschiedliche Ausprägungen digitaler Medien vergleichen und die jeweiligen medialen Mehrwerte bestimmen. Dies ist für alle Anwendungsbereiche, die heutzutage mit dem E-Präfix versehen sind (E-Learning, E-Government, E-Science, E-Business etc.) von entscheidender Bedeutung.

Literaturangaben

Werden in der Veranstaltung bekannt gegeben.

Verschiedenes

Hörerkreis:

i-l, i-m, ie, ii, im, winf

Prüfungsform:

Gruppenarbeit mit Jour Fixe und abschließende mündliche Einzelüberprüfung; regelmäßige Teilnahme erforderlich!

weiterführende Veranstaltungen:

siehe Modulhandbuch

Homepage:

koala.uni-paderborn.de

Modulzugehörigkeit:

III.4.2 Informatik und Gesellschaft, III.4.4 Computergestütztes kooperatives Arbeiten und Lernen

vorausgesetzte Kenntnisse:

Grundkenntnisse aus den ersten vier Semestern

nächster Wiederholungstermin:

WS 17/18

Einführung in Informatik und Gesellschaft

Dozent: Selke

Büro: F1.101

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Informatiker entwickeln auf Zeichen basierende Produkte (Programme, Spezifikationen, Dokumentationen etc.). Im Gegensatz zu anderen Ingenieurprodukten, die aus Materialien wie Stahl, Kunststoff oder Glas gefertigt werden, bildet Software soziale Wirklichkeit in vielfältiger Art und Weise ab. Durch den Einsatz ändert sich diese Wirklichkeit. Das führt zu Wechselwirkungen zwischen Informatiksystemen und ihrem Einsatzumfeld; durch den Einsatz der Systeme verändert sich das Verhalten der Menschen, das in den Systemen modelliert oder in Form von Annahmen verankert ist. Das Einsatzumfeld wirkt auf das Produkt zurück – Revisionen, Anpassungen und Erweiterungen sind die Folge. Diese Wechselwirkungen gilt es möglichst frühzeitig zu erkennen, um Gefahren abwenden und zukünftige Anpassungen antizipieren zu können.

Ziel ist es, das Zusammenspiel spezifischer Technologien mit kognitiven, sozialen, ökonomischen und politischen Faktoren zu untersuchen und dabei sowohl Risiken als auch Chancen zu identifizieren. Die Vorlesung „Einführung in Informatik und Gesellschaft“ legt dazu die theoretischen und begrifflichen Grundlagen, die in weiteren Veranstaltungen des Moduls ergänzt bzw. vertieft werden.

Literaturangaben

Wird in der Veranstaltung bekanntgegeben.

Verschiedenes

Hörerkreis:

i-I, i-m, ie, ii, im, winf

Prüfungsform:

Modulprüfung

weiterführende Veranstaltungen:

Assistierende Technologien, Barrierefreiheit
Seminare aus dem Gebiet Mensch-Maschine-
Wechselwirkung

Vorbesprechung:

keine

Modulzugehörigkeit:

III.4.2 Informatik und Gesellschaft

vorausgesetzte Kenntnisse:

Bachelor in Informatik oder vergleichbare
Kenntnisse

nächster Wiederholungstermin:

WS 17/18

Homepage:

koala.uni-paderborn.de

Seminar Formal Models for Safe and Secure Systems

Dozent: Wehrheim

Büro: O4.225

Sprechstunde: nach Vereinbarung

Inhaltsangabe

In this seminar, we will study different formal modelling techniques. All techniques will be based on some sort of automata. Such models can be employed in a formal approach to system correctness, both with respect to safety as well as security properties.

The models we study will most likely include automata over infinite words, probabilistic, timed and hybrid automata, as well as security automata and IO-automata. Some of the presentations will also describe automata-based verification techniques.

Verschiedenes

Hörerkreis:

Master Computer Science

Prüfungsform:

Presentation + report + active participation
+ reviews of other reports

Vorbesprechung:

to be announced, see webpages of research group

Modulzugehörigkeit:

SWT & IS, Module 1.1 und 1.5

vorausgesetzte Kenntnisse:

either course Model Checking or course Deductive Verification,
logic and finite automata

Seminar: Secure Systems Engineering (in English)

Dozent: Bodden

Büro: ZM1.02-09

Sprechstunde: Nach Absprache

Inhaltsangabe

This seminar will cover various topics in Software and Systems Engineering as well as Software and Systems Security.

Students are expected to prepare a seminar paper, to deliver a presentation, and to participate in a peer-review process of other students' seminar papers.

There will be an initial meeting in which seminar topics are announced and then allocated. Afterwards the students/groups meet mostly 1-on-1 with their topic supervisor. In the end there will be a block seminar in which presentations are delivered.

Verschiedenes

Hörerkreis:

Informatik Master

Modulzugehörigkeit:

M.079.3115 Analytische Methoden des Software Engineering (2009), M.079.3116 Konstruktive Methoden des Software Engineering (2009), M.079.3111 Modellbasierte Softwareentwicklung (2009) , M.079.3112 Sprachen und Programmiermethoden (2009)

Prüfungsform:

Seminar paper and Presentation

Leistungspunkte:

4

nächster Wiederholungstermin:

WS 2017/18

Vorbesprechung:

TBA

Homepage:

<https://www.hni.uni-paderborn.de/swt/lehre/secure-systems-engineering/>

Seminar: Gems of Theoretical Computer Science

Dozent: Meyer auf der Heide

Büro: F1.301

Sprechstunde: Mi, 13-14 Uhr

Inhaltsangabe

The seminar based on a set of selected papers and textbook sections that demonstrate the beauty of problem solutions in the field of Theoretical Computer Science. We will find out that the preoccupation with sophisticated proof techniques, elegant arguments and surprising constructions is highly enjoyable. The seminar is inspired by the book “Perlen der Theoretischen Informatik” by Uwe Schöning, in which he presents a collection of results demonstrating highlights of Theoretical Computer Science. Of course, the selection of seminar topics is affected by the supervisor’s taste and research area. The seminar is expected to take place as a block seminar at the end of the lecture period.

Literaturangaben

- **Uwe Schöning:** *Perlen der Theoretischen Informatik*, Spektrum Akademischer Verlag, 2002

Verschiedenes

Hörerkreis:

Master Informatik

Prüfungsform:

seminar thesis, presentation

Vorbesprechung:

first week of lectures, to be announced on
lecture’s web site

Modulzugehörigkeit:

Seminar: Modul III 2.1, 2.2, 2.3, 2.4

nächster Wiederholungstermin:

Winter Term 2017/18

Homepage:

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

Seminar Complexity and Cryptography

Dozent: Blömer

Büro: F2.101

Sprechstunde: Mi, 11-12 Uhr

Inhaltsangabe

We will discuss current projects from the area of complexity and cryptography. Some of the topics covered are:

- securely outsourcing computation: Yao's garbled circuits
- securely outsourcing storage: oblivious RAM
- useful but insecure: attacks on order-preserving encryption
- anonymizing large datasets: differential privacy
- distributed online poker without a trusted dealer
- proving knowledge of a number in a certain range
- transforming sigma protocols to perfect zero-knowledge protocols

Literaturangaben

To be discussed in the first meeting during the first week of lectures.

Verschiedenes

Hörerkreis:

Master Informatik

Prüfungsform:

Vortrag, Ausarbeitung

Vorbesprechung:

Vorbesprechung in der ersten Semesterwoche

Modulzugehörigkeit:

Mastermodul III.2.3; Mastermodul III.2.2

vorausgesetzte Kenntnisse:

Mastervorlesungen im Bereich Komplexitätstheorie oder Kryptographie

Master Seminar on Selected Topics in Wireless Networking

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:

Sommer 2017

Vorbesprechung:

erste Semesterwoche

Homepage:

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

Seminar: Geschichte interaktiver Systeme

Dozent: Selke

Büro: F1.101

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Bei der Gestaltung interaktiver Medien geht es darum, Systeme zu entwickeln, die intelligentes Verhalten der Nutzer unterstützen bzw. verstärken, nicht aber ersetzen wollen. Unter der Maxime „Augmenting the Human Intellect“ wurden Techniken wie Hypertext, Objektorientierte Grafik, Interreferentielle Ein-/Ausgabe, Fenstersysteme, virtuelle Realität, etc. entwickelt. Im Proseminar stellen die Teilnehmer die wesentlichen Meilensteine dieser Entwicklung vor, analysieren entsprechende Konzepte und bewerten den Einfluss auf zukünftige Entwicklungen.

Das Seminar findet als Blockveranstaltung am Ende des Wintersemesters statt. Die Themenvergabe findet bei der Vorbesprechung statt.

Literaturangaben

Werden in der Veranstaltung bekannt gegeben!

Verschiedenes

Modulzugehörigkeit:

beliebiges Modul im Bereich MMWW

nächster Wiederholungstermin:

WS 17/18

Homepage:

koala.uni-paderborn.de

Prüfungsform:

Vortrag und Ausarbeitung

Vorbesprechung:

Mi, 19. Oktober 2016, 14 Uhr, F1.544

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

Prüfungsform:

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

weiterführende Veranstaltungen:

Master Thesis

Modulzugehörigkeit:

III.5.1 Projekt Group

vorausgesetzte Kenntnisse:

Bachelor Computer Science, Java, Eclipse Plugin Development

Homepage:

-

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 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

nächster Wiederholungstermin:

SS2017

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)

Project Group on Wireless Networking

Dozent: Dressler

Büro: F1.401

Sprechstunde: nach Absprache

Inhaltsangabe

This project group is being offered to complement our “Master Seminar” series, each issue of which treats current topics in the context of active research domains.

Verschiedenes

Hörerkreis:

Master Informatik, Master Computer Engineering

Homepage:

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

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

nächster Wiederholungstermin:

SS 2017

vorausgesetzte Kenntnisse:

Fortgeschrittenes Wissen im Bereich der theoretischen Informatik.

Homepage:

<http://cs.uni-paderborn.de/ti/lehre/oberseminar/ws-2016/index.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>

Oberseminar Rechnernetze

Dozent: Karl

Büro: O3.158

Sprechstunde: Montag, 13 - 14 Uhr

Inhaltsangabe

Verschiedene Vorträge im Bereich Rechnernetze.

Verschiedenes

Hörerkreis:

Bachelor, Master

Modulzugehörigkeit:

ESS

nächster Wiederholungstermin:

SS 2017

Grundlagen der Programmierung für Ingenieure II

Dozent: Fischer

Büro: F1.223

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Veranstaltung richtet sich an Erstsemesterstudierende der Studienrichtungen Elektrotechnik.

Das Hauptziel der Veranstaltung ist die Vermittlung von Grundkenntnissen einer Programmiersprache C++. Dazu gehören einfache Anweisungen, Ablaufsteuerungen, Arrays, dynamische Datenstrukturen und objektorientierte Programmierung. Die Vermittlung dieser Grundkenntnisse wird von Übungen am Computer begleitet.

Für Studierende der Informatik, Wirtschaftsinformatik, Ingenieurinformatik usw., die die Veranstaltung im Rahmen des Studium Generale belegen möchten, wird keine Prüfungsmöglichkeit angeboten. Das gilt auch für Studierende der Mathematik. Mathematik Studierende seien auf die vergleichbare Veranstaltung der Mathematik „C-Programmierung Mathematik“ verwiesen. Für informatikferne Fächer wie Physik, Medienwissenschaften, Kulturwissenschaften usw. ist die Möglichkeit zur Prüfung im Rahmen Studium Generale gegeben. Im Zweifel kontaktieren Sie den Veranstalter.

Literaturangaben

- **Ulrich Breymann:** *C++ Einführung und professionelle Programmierung*,
- **Bjarne Stroustrup:** *The C++ Programming Language*,
- **Bjarne Stroustrup, Petra Alm und Dirk Louis:** *Einführung in die Programmierung mit C++*,

Verschiedenes

Hörerkreis:

1. Semester Maschinenbau, Wirtschaftsingenieurwesen, Elektrotechnik

Modulzugehörigkeit:

Grundstudium

Prüfungsform:

Klausur

vorausgesetzte Kenntnisse:

keine

nützliche Parallelveranstaltungen:

keine

nächster Wiederholungstermin:

WS 2017/18

Vorbesprechung:

keine

Homepage:

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

Grundlagen der Programmierung für MB

Dozent: Fischer

Büro: F1.223

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Veranstaltung richtet sich an Erstsemesterstudierende der Studienrichtungen Maschinenbau, Wirtschaftsingenieurwesen und Elektrotechnik.

Das Hauptziel der Veranstaltung ist die Vermittlung von Grundkenntnissen einer Programmiersprache. Ab dem WS 12/13 ist das die Sprache C++. Dazu gehören einfache Anweisungen, Ablaufsteuerungen, Arrays, dynamische Datenstrukturen und objektorientierte Programmierung. Die Vermittlung dieser Grundkenntnisse wird von Übungen am Computer begleitet.

Für Studierende der Informatik, Wirtschaftsinformatik, Ingenieurinformatik usw., die die Veranstaltung im Rahmen des Studium Generale belegen möchten, wird keine Prüfungsmöglichkeit angeboten. Das gilt auch für Studierende der Mathematik. Mathematik Studierende seien auf die vergleichbare Veranstaltung der Mathematik „C-Programmierung Mathematik“ verwiesen. Für informatikferne Fächer wie Physik, Medienwissenschaften, Kulturwissenschaften usw. ist die Möglichkeit zur Prüfung im Rahmen Studium Generale gegeben. Im Zweifel kontaktieren Sie den Veranstalter.

Literaturangaben

- **Ulrich Breymann:** *C++ Einführung und professionelle Programmierung*,
- **Bjarne Stroustrup:** *The C++ Programming Language*,
- **Bjarne Stroustrup, Petra Alm und Dirk Louis:** *Einführung in die Programmierung mit C++*,

Verschiedenes

Hörerkreis:

1. Semester Maschinenbau, Wirtschaftsingenieurwesen, Elektrotechnik

Modulzugehörigkeit:

Grundstudium

Prüfungsform:

Klausur

vorausgesetzte Kenntnisse:

keine

nützliche Parallelveranstaltungen:

keine

nächster Wiederholungstermin:

WS 2017/18

Vorbesprechung:

keine

Homepage:

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

Einführung in die Web-bezogenen Sprachen

Dozent: Pfahler

Büro: F2.311

Sprechstunde: Nach Vereinbarung

Inhaltsangabe

Diese Vorlesung wird für Studierende des Studienganges Medienwissenschaften angeboten.

Ziele:

Die Vorlesung soll Studierende dazu befähigen,

- Sprachen, die zur Entwicklung von Web-Präsenzen eingesetzt werden, zu verstehen, anzuwenden und zu beurteilen,
- einfache Web-Präsenzen mit den dafür heute gebräuchlichen Sprachen und Methoden zu entwickeln,
- Sprachen, die in Zukunft für solche Aufgaben eingesetzt werden, dann selbständig zu erlernen,
- grundlegende, allgemeine Programmier Techniken anzuwenden.

Verschiedenes

Hörerkreis:

MeWi

Leistungspunkte:

8

nächster Wiederholungstermin:

Winter 2017/2018

Prüfungsform:

Klausur

vorausgesetzte Kenntnisse:

Einführung in die Informatik für Geisteswissenschaften

Homepage:

<http://homepages.uni-paderborn.de/peter/ews/ews.html>

Datenverarbeitung

Dozent: Fischer

Büro: F1.223

Sprechstunde: nach Vereinbarung

Inhaltsangabe

Die Veranstaltung richtet sich an Erstsemesterstudierende der Studienrichtungen Elektrotechnik und Wirtschaftsingenieurwesen.

Das Hauptziel der Veranstaltung ist die Vermittlung von Grundkenntnissen einer Programmiersprache C++. Dazu gehören einfache Anweisungen, Ablaufsteuerungen, Arrays, dynamische Datenstrukturen und objektorientierte Programmierung. Die Vermittlung dieser Grundkenntnisse wird von Übungen am Computer begleitet.

Für Studierende der Informatik, Wirtschaftsinformatik, Ingenieurinformatik usw., die die Veranstaltung im Rahmen des Studium Generale belegen möchten, wird keine Prüfungsmöglichkeit angeboten. Das gilt auch für Studierende der Mathematik. Mathematik Studierende seien auf die vergleichbare Veranstaltung der Mathematik „C-Programmierung Mathematik“ verwiesen. Für informatikferne Fächer wie Physik, Medienwissenschaften, Kulturwissenschaften usw. ist die Möglichkeit zur Prüfung im Rahmen Studium Generale gegeben. Im Zweifel kontaktieren Sie den Veranstalter.

Literaturangaben

- **Ulrich Breymann:** *C++ Einführung und professionelle Programmierung*,
- **Bjarne Stroustrup:** *The C++ Programming Language*,
- **Bjarne Stroustrup, Petra Alm und Dirk Louis:** *Einführung in die Programmierung mit C++*,

Verschiedenes

Hörerkreis:

1. Semester Maschinenbau, Wirtschaftsingenieurwesen, Elektrotechnik

Modulzugehörigkeit:

Grundstudium

Prüfungsform:

Klausur

vorausgesetzte Kenntnisse:

keine

nützliche Parallelveranstaltungen:

keine

nächster Wiederholungstermin:

WS 2017/18

Vorbesprechung:

keine

Homepage:

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

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					