

Vorlesungskommentar

Wintersemester 2007/2008



Fachbereich Informatik

Universität Dortmund

VERANSTALTUNGEN IM GRUNDSTUDIUM	5
040101 <i>Datenstrukturen, Algorithmen und Programmierung I</i>	5
040102 <i>Übung zu Datenstrukturen, Algorithmen und Programmierung I</i>	6
040103 <i>Rechnerstrukturen</i>	6
040104 <i>Übung zu Rechnerstrukturen</i>	7
040301 <i>Softwaretechnik</i>	7
040302 <i>Übung zu Softwaretechnik</i>	8
040303 <i>Betriebssysteme, Rechnernetze und verteilte Systeme II</i>	8
040304 <i>Übung zu Betriebssysteme, Rechnernetze und verteilte Systeme II</i>	9
010116 <i>Logik für Informatik</i>	9
010117 <i>Übung zu Logik für Informatik</i>	10
040305 <i>Funktionale Programmierung</i>	10
040306 <i>Übung zu Funktionale Programmierung</i>	10
KURS	11
040505 <i>Programmierkurs für Umsteiger (Wechsel von Diplom- in Bachelorstudiengänge)</i>	11
PRAKTIKA	12
040501 <i>Hardwarepraktikum für Informatiker</i>	12
040503 <i>Programmierpraktikum DAP 1</i>	12
040508 <i>Softwarepraktikum in der vorlesungsfreien Zeit</i>	12
PROSEMINARE	14
040727 <i>Meilensteine der Theoretischen Informatik</i>	14
040728 <i>Algorithmen für NP- vollständige Probleme</i>	14
VERANSTALTUNGEN IM HAUPTSTUDIUM	16
PFLICHTVERANSTALTUNGEN	16
041025 <i>Übersetzerbau</i>	16
041026 <i>Übung zu Übersetzerbau</i>	16
041030 <i>Unternehmensgründung (Informatik im Kontext)</i>	16
041031 <i>Projektmanagement aus der Praxis (Informatik im Kontext)</i>	17
041032 <i>Übung zu Projektmanagement aus der Praxis (Informatik im Kontext)</i>	18
WAHLPFLICHTVERANSTALTUNGEN	19
041121 <i>Mensch-Maschine-Interaktion</i>	19
041122 <i>Übung zu Mensch-Maschine-Interaktion</i>	20
041125 <i>Eingebettete Systeme</i>	20
041126 <i>Übung zu Eingebettete Systeme</i>	21
041223 <i>Darstellung, Verarbeitung und Erwerb von Wissen</i>	21
041224 <i>Übung zu Darstellung, Verarbeitung und Erwerb von Wissen</i>	22
041221 <i>Komplexitätstheorie und Effiziente Algorithmen</i>	22
041222 <i>Übung zu Komplexitätstheorie und Effiziente Algorithmen (KT+EA)</i>	23
042381 <i>Formale Methoden des Systementwurfs 2</i>	24
042382 <i>Übung zu Formale Methoden des Systementwurfs 2</i>	24
WAHLVERANSTALTUNGEN	25
042397 <i>Mustererkennung</i>	25
042398 <i>Übung zu Mustererkennung</i>	25
042417 <i>Praktische Optimierung</i>	26
048418 <i>Übung zu Praktische Optimierung</i>	26
042379 <i>Randomisierte Algorithmen</i>	26
042380 <i>Übung zu Randomisierte Algorithmen</i>	27
042371 <i>Petrinetze</i>	27
042373 <i>Desktop Video</i>	28
042375 <i>Übersetzerbau 2</i>	29
042377 <i>Logisch-algebraischer Systementwurf 1</i>	29
042383 <i>SOS und Views II</i>	29
042385 <i>Geometrisches Modellieren</i>	30
042386 <i>Übung zu Geometrisches Modellieren</i>	30

042387	Datenvisualisierung	31
042388	Übung zu Datenvisualisierung	31
042391	Automatisches Zeichnen von Graphen	32
042392	Übung zu Automatisches Zeichnen von Graphen	32
042395	Computer Vision	33
042396	Übung zu Computer Vision	33
042397	Mustererkennung	33
042398	Übung zu Mustererkennung	34
042399	Workflows in Theorie und Praxis	34
042401	Modellierung und Simulation diskreter und kontinuierlicher Systeme	34
042402	Übung zu Modellierung und Simulation diskreter und kontinuierlicher Systeme	35
042405	Betriebssystembau	35
042406	Übungen zu Betriebssystembau	36
042407	Quantenalgorithmen und Quantenkryptographie	36
042409	Compiler für Eingebettete Systeme	37
042411	Reaktive Sicherheit	38
042412	Übung zu Reaktive Sicherheit	39
042413	Kommunikationskomplexität	39
042415	Medienengineering	40
042416	Übung zu Medienengineering	40
SEMINARE		41
044651	Executable Specification Languages	41
044652	Intelligente Anwendungen im Internet	41
044653	Maschinelles Lernen und Wissensentdeckung	42
044654	Document Engineering	42
044655	Algorithm Engineering	43
044656	Relationales Lernen und Modellieren	44
044657	Eingebettete Systeme	44
044658	Simulated Annealing	44
044659	Algorithmische Spieltheorie	45
044660	Suchmaschinen	46
044661	Mobile Learning 2	46
VERANSTALTUNGEN FÜR LEHRAMTSTUDIERENDE		47
041301	Einführung in die Didaktik der Informatik	47
041309	Grundlagen der Oberstufeninformatik	47
BESONDERE VERANSTALTUNGEN		49
049991	Kolloquium des Fachbereichs Informatik	49
049990	CI-Kolloquium	49
VERANSTALTUNGEN DER RUHR-UNIVERSITÄT BOCHUM FÜR STUDIERENDE MIT NEBENFACH 'THEORETISCHE MEDIZIN'		50
LEHRVERANSTALTUNGEN VOR DEM VORDIPLOM		50
209500	Anatomie I	50
209501	Biochemie I	50
209502	Physiologie I	50
LEHRVERANSTALTUNGEN NACH DEM VORDIPLOM		50
206003	Hygiene	50
209503	Pathologie	51
ANHANG		52
LEHRANGEBOT ZU DEN BACHELOR- UND MASTERSTUDIENGÄNGEN INFORMATIK UND ANGEWANDTE INFORMATIK		
INFORMATIONEN ZUM WECHSEL VON DEN DIPLOMSTUDIENGÄNGEN (NACH DPO 03) IN DIE BACHELORSTUDIENGÄNGE		
BEISPIELHAFTER ZEITLICHER VERLAUF EINES INFORMATIK-STUDIUMS NACH DPO 01		
VORLESUNGSZYKLEN		
ZUORDNUNG LEHRVERANSTALTUNGEN ZU SCHWERPUNKTGEBIETEN		

**PRÜFUNGSGBIETE DER HOCHSCHULLEHRER DES FACHBEREICHS INFORMATIK
(ZZGL. EIGENE VORLESUNGEN)**

SPRECHZEITEN PROFESSORINNEN/PROFESSOREN, HABILITIERTE

WICHTIGE SPRECHZEITEN WS 2007/2008

STUDIENFACHBERATUNG DES FACHBEREICHS INFORMATIK

Die Zwischenüberschriften beziehen sich auf die Diplomstudiengänge. Für die Zuordnung der Lehrveranstaltungen zu den Bachelor-/Master-Studiengängen siehe Tabellen im Anhang.

Veranstaltungen im Grundstudium

040101 **Datenstrukturen, Algorithmen und Programmierung I**

Doberkat, Ernst-Erich

Vorlesung

4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	12:15	14:00	wöchentlich	HG III / Audimax
	Donnerstag	14:15	16:00	wöchentlich	HG II / HS 1
Kommentar	Diese Veranstaltung, die neben einer Vorlesung auch Übungen umfaßt, und die eng mit dem Programmierpraktikum I gekoppelt ist, ist die erste Veranstaltung im Software-Zyklus und fundamental für das weitere Informatik-Studium. In ihr werden mit der Programmiersprache JAVA Konzepte für die strukturierte und objektorientierte Programmierung konzeptionell und praktisch vermittelt, es findet ebenfalls eine informelle, exemplarische Diskussion von Syntax und Semantik einer Programmiersprache statt. Eine erste, exemplarische Einführung in die systematische Konstruktion von Algorithmen diskutiert u. a. diese Themen: Sortieren auf Feldern, Verwalten von Listen, Verwalten und Traversieren von Bäumen mit verschiedenen Strategien, Suchen und Sortieren mit Bäumen; elementare Algorithmen auf Graphen, eng verwandt damit ist ein erster Einblick in einige klassischen Datenstrukturen wie Felder, Listen, Bäume, Graphen und implizite Datenstrukturen wie Heaps. Das gegenwärtig vorherrschende Programmierparadigma ist die objektorientierte Software-Konstruktion, hier werden die folgenden Themen behandelt: Geheimnisprinzip und Kapselung bei der Konstruktion von Klassen, Nachrichtenaustausch zwischen Objekten, Vererbung, Aufbau von Spezialisierungshierarchien und Abbildung auf Vererbungshierarchien, Einsatz von Ausnahmebehandlung, Anwendung von Generizität; die Veranstaltung schließt mit der Besprechung einfacher Entwurfsmuster und des objektorientierten Entwurfs, so dass eine Brücke zum Modul Software-Technik geschlagen wird. Die begleitenden Übungen zu DAP 1 dienen zur Vertiefung des in der Vorlesung behandelten Stoffes. Dies geschieht durch regelmäßig ausgegebene Übungsaufgaben, die die Studenten selbstständig bearbeiten. In den Präsenzzeiten der Übung werden die Lösungen der Aufgaben in kleineren Übungsgruppen besprochen. Nach erfolgreichem Abschluss sollen die Teilnehmer die informellen Grundlagen für die Beschreibung von Programmiersprachen und exemplarisch deren Umsetzung im Rahmen der Programmiersprache JAVA kennen und einordnen. Die Studierenden beherrschen die Grundbegriffe der objektorientierten Programmgestaltung, sie können für gegebene Problemstellungen selbstständig Lösungsalgorithmen formulieren und diese als JAVA-Programme praktisch implementieren. Sie beherrschen ausgewählte Entwurfsmuster für die objektorientierte Softwarekonstruktion und können ihre Verwendbarkeit einschätzen. Das zugehörige Programmierpraktikum legt seinen Schwerpunkt auf die praktische Realisierung der entsprechenden JAVA-Programme, die behandelten Themen durch Arbeiten am Computer praktisch eingeübt.				

Literatur E.-E. Doberkat, S. Dißmann: Obektorientierte Programmierung mit JAVA, Oldenbourg-Verlag, München, 2. Auflage, 2002

040102 Übung zu Datenstrukturen, Algorithmen und Programmierung I

Dißmann, Stefan ; Eskin, Esmeray ; Menge, Sebastian ; Rupflin, Wilfried ; Sugioarto, Martin

Übung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Freitag	08:15	10:00	wöchentlich	GB IV / 126
	Freitag	08:15	10:00	wöchentlich	GB IV / 113
	Freitag	08:15	10:00	wöchentlich	GB IV / 228
	Freitag	08:15	10:00	wöchentlich	GB IV / 318
	Freitag	08:15	10:00	wöchentlich	GB V / 420
	Freitag	10:15	12:00	wöchentlich	GB V / 420
	Freitag	10:15	12:00	wöchentlich	GB IV / 318
	Freitag	10:15	12:00	wöchentlich	GB IV / 228
	Freitag	10:15	12:00	wöchentlich	GB IV / 126
	Freitag	10:15	12:00	wöchentlich	GB IV / 113
	Freitag	12:15	14:00	wöchentlich	GB V / 420
	Freitag	12:15	14:00	wöchentlich	GB IV / 113
	Freitag	12:15	14:00	wöchentlich	GB IV / 126
	Freitag	12:15	14:00	wöchentlich	GB IV / 318
	Freitag	12:15	14:00	wöchentlich	GB IV / 228
	Freitag	14:15	16:00	wöchentlich	GB IV / 126
	Freitag	14:15	16:00	wöchentlich	GB IV / 318
	Freitag	14:15	16:00	wöchentlich	GB IV / 228
	Freitag	14:15	16:00	wöchentlich	GB IV / 113
	Freitag	14:15	16:00	wöchentlich	GB V / 420

040103 Rechnerstrukturen

Jansen, Thomas ; Marwedel, Peter

Vorlesung 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	16:15	18:00	wöchentlich	HG III / Audimax
	Mittwoch	12:15	14:00	wöchentlich	EF 50 / HS 1

Kommentar In der Vorlesung "Rechnerstrukturen" werden die Grundlagen der technischen Realisierung von Anlagen zur Informationsverarbeitung und deren Schnittstelle zur Software thematisiert. Es geht also grundsätzlich um Hardware und deren Einsatz zur Ausführung von Software, im Englischen auch als "execution platforms" bezeichnet. Wir beschreiben Sichten auf Rechenanlagen, die verschiedenen Abstraktionsebenen entsprechen. Aus dieser Perspektive heraus werden wir uns mit folgenden Inhalten befassen:

- Darstellung von Informationen in Rechnern
- Boolesche Funktionen, Repräsentationen und Realisierung
- Schaltnetze
- endliche Automaten
- Schaltwerke
- Hardware-Komponenten zur Realisierung von Prozessoren,
- die Befehlsschnittstelle,

- Assemblerprogrammierung,
- exemplarische Realisierung eines Prozessors aus Hardware-Komponenten,
- die Speicherarchitektur,
- Sekundärspeicher,
- Ein-/Ausgabeorganisation,
- Anwendungen von Rechnern in eingebetteten Systemen.

Weitere Angaben sind auch im Modulhandbuch für die Bachelor-Studiengänge zu finden: <http://dekanat.cs.uni-dortmund.de/Studium/modulkataloge.html>

Literatur

Literatur:

Die Vorlesung orientiert sich an dem gleichnamigen Skriptum in zwei Teilen (Autor Teil 1: Th. Jansen; Autor Teil 2: P. Marwedel). Auf der Erstsemester-CD wie auch im Web werden die beiden Skripte und teilweise Folien zur Verfügung gestellt. Die CD enthält auch den SPIM-Simulator, mit dem die Assemblerprogrammierung praktisch erprobt werden wird. Außerdem enthält die CD multimediale Lehreinheiten des Ra-Vi-Projekts, mit denen das dynamische Verhalten von Rechensystemen visualisiert werden kann.

Hinweise auf zusätzliche und weiterführende Literatur zum ersten Teil findet man sowohl im Skriptum zur Vorlesung als auch auf der entsprechenden Webseite. Zum zweiten Teil der Vorlesung sollte ergänzend zum Skriptum das Buch J. Hennessy, D. Patterson: Computer Organization and Design: The hardware/software interface (siehe Lehrbuchsammlung Informatik) benutzt werden. Weitere nützliche Informationen findet man in dem Buch "Mikrorechner-Technik" von H. Bähring (siehe ebenfalls Lehrbuchsammlung Informatik).

Bemerkungen

Der Besuch dieser Vorlesung sollte für das erste Semester eingeplant werden.

Es finden Übungen zur Vorlesung statt. Zur Vorbereitung auf die Klausur am Ende der Vorlesungszeit wird die aktive Teilnahme an den Übungen dringend empfohlen.

Leistungsnachweis

Bei Bestehen der Klausur werden 7,5 Leistungspunkte vergeben.

040104 **Übung zu Rechnerstrukturen**

Jansen, Winfried ; Müller, Dennis ; Thimm, Matthias ; Wurst, Michael ; Zarges, Christine

Übung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	10:15	12:00	wöchentlich	GB IV / 126
	Montag	10:15	12:00	wöchentlich	GB IV / 228
	Montag	12:15	14:00	wöchentlich	GB IV / 126
	Montag	12:15	14:00	wöchentlich	GB IV / 318
	Montag	12:15	14:00	wöchentlich	GB IV / 228
	Dienstag	08:15	10:00	wöchentlich	GB IV / 126
	Dienstag	10:15	12:00	wöchentlich	GB IV / 126
	Dienstag	10:15	12:00	wöchentlich	GB IV / 228
	Mittwoch	10:15	12:00	wöchentlich	GB IV / 113
	Mittwoch	10:15	12:00	wöchentlich	GB IV / 126
	Donnerstag	10:15	12:00	wöchentlich	GB IV / 126
	Donnerstag	10:15	12:00	wöchentlich	GB IV / 228

040301 **Softwaretechnik**

Fronk, Alexander

Vorlesung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	14:15	16:00	wöchentlich	HG I / HS 6
	Freitag	12:15	14:00	wöchentlich	HG I / HS 6
Voraussetzungen	Die Veranstaltung richtet sich an Hörerinnen und Hörer in dritten Fachsemester Informatik oder Angewandte Informatik. Es wird angenommen, dass die Inhalte der Veranstaltungen DAP 1/2 nicht nur bekannt, sondern auch vertraut sind; das gilt ebenfalls für die Programmiersprache Java. Diese Vorlesung ist das Bindeglied zwischen dem DAP-Zyklus und dem Software-Praktikum; die Teilnahme am SoPra setzt die erfolgreiche Teilnahme an der Vorlesung Softwaretechnik voraus.				
Kommentar	Die Modellierung ist eine wesentliche Tätigkeit in der Softwaretechnik, die durch eine Vielzahl von Formalismen unterstützt wird. In dieser Vorlesung soll gezeigt werden, wie mit Hilfe der Unified Modelling Language (UML) Modelle erarbeitet und realisiert werden können. Dazu wird diese graphisch orientierte Sprache eingeführt und in ihren vielseitigen Aspekten diskutiert. Es wird gezeigt, in welchen Phasen des Konstruktionsprozesses von Software welche Modellkomponenten benutzt werden, welche Eigenschaften die Diagramme haben und welche Grenzen ihrer Verwendung gesetzt sind.				
Literatur	wird noch bekannt gegeben.				
Bemerkungen	Die Veranstaltung wird in diesem Jahr 4-stündig gehalten, sie endet jedoch mit den Weihnachtsferien. Entsprechend finden die Übungen 2-stündig statt. Die Klausur wird in der letzten Vorlesungswoche vor den Weihnachtsferien stattfinden, die Nachklausur am Ende der Vorlesungszeit.				
Leistungsnachweis	Es wird ein Schein vergeben, falls die Klausur bzw. die Nachklausur mit der gemäß beim ersten Veranstaltungstermin bekannt gegebenen Punktesumme bestanden wird. Eine Teilnahme an der Nachklausur ist nur möglich, wenn die Klausur nicht bestanden wurde.				

040302 Übung zu Softwaretechnik

Doedt, Markus ; Wiese, Lena

Übung

1 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	10:15	12:00	wöchentlich	GB V / 420
	Mittwoch	12:15	14:00	wöchentlich	GB IV / 318
	Donnerstag	08:15	10:00	wöchentlich	GB V / 324
	Donnerstag	10:15	12:00	wöchentlich	GB V / 324

040303 Betriebssysteme, Rechnernetze und verteilte Systeme II

Buchholz, Peter

Vorlesung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	12:15	14:00	wöchentlich	HG II / HS 3
Voraussetzungen	Veranstaltung BSRvS1 und Programmierkenntnisse in JAVA.				

- Kommentar** Rechnerhardware und Systemsoftware bilden zusammen eine integrale Plattform, über die es erst möglich wird, Anwendungssoftware auszuführen. Der Vorlesungszyklus RvSBS 1/2 widmet sich diesem Themenfeld und soll in die Grundlagen und Architekturkonzepte von Betriebssystemen, Kommunikationssystemen, Middleware-Plattformen und anderen Unterstützungssystemen für verteilte Anwendungen einführen.
Die Vorlesung RvSBS2 konzentriert sich dazu auf den Themenbereich "Rechnernetze" mit den beiden Schwerpunkten "Telekommunikation" und "Verteilte Verarbeitung". Ein Schwerpunkt behandelt, wie mit Hilfe einer Hierarchie aus Kommunikationsdiensten und Protokollen Datennetze gebildet werden. Über ein Datennetz verbundene Rechner bilden ein Rechnernetz. Darüber hinaus werden spezifische Aspekte aus dem Rechnernetzbereich, wie die Verbreitung von Multimediadaten im Netz, die Sicherheit in vernetzten Umgebungen und das Management von Netzen behandelt.
- Literatur** Basis der Vorlesung ist das Buch
J. F. Kurose and K. W. Ross: Computer Networking: A Top-Down Approach. Addison Wesley 2008
oder die ältere deutsche Version
J. F. Kurose and K. W. Ross: Computernetze: Ein Top-Down-Ansatz mit Schwerpunkt Internet. Pearson Studium 2002.
- Die Folien und Übungen werden in elektronischer Form verfügbar gemacht.

040304 Übung zu Betriebssysteme, Rechnernetze und verteilte Systeme II

Arns, Markus ; Tатаh, Veye Wirngo

Übung 1 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	12:15	14:00	wöchentlich	GB V / 324
	Montag	12:15	14:00	wöchentlich	GB V / 420
	Dienstag	12:15	14:00	wöchentlich	GB IV / 113
	Dienstag	12:15	14:00	wöchentlich	GB IV / 228

- Kommentar** Es werden wie in "BS+RvS 1" Übungen veranstaltet, die nicht auf schriftlichen Ausarbeitungen beruhen, sondern in kleinen Gruppen die Vorlesungsinhalte durch Diskussion und Aktivitäten wie Rollenspielen vertieft werden sollen. Die Teilnahme sollte im Hinblick auf die besondere Relevanz für die Klausur als Pflicht betrachtet werden. Die Anmeldung erfolgt über Templus.

010116 Logik für Informatik

Schwentick, Thomas

Vorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	10:00	12:00	wöchentlich	HG II / HS 1

- Kommentar** Die Veranstaltung führt in die moderne mathematische Logik ein. Neben klassischen logischen Systemen wie der Aussagenlogik und der Prädikatenlogik lernen die Studierenden diverse moderne logische Systeme kennen, die in der Informatik verwendet werden. Außer der Syntax und der Semantik dieser logischen Systeme werden verschiedene Kalküle behandelt, mit deren Hilfe man die Unerfüllbarkeit logischer Formeln untersuchen kann.
Es soll die Fähigkeit vermittelt werden, für die Modellierung einer gegebenen Problemstellung ein geeignetes logisches System zu finden bzw. zu entwickeln. Die

Studierenden sollen Syntax und Semantik verschiedener logischer Systeme beherrschen und ggf. weiter entwickeln können. Sie sollen auch eine Reihe klassischer Kalküle kennen und diese auf konkrete Problemstellungen anwenden können.

Die Vorlesung wird im Wesentlichen dem Buch Logik für Informatiker von Martin Kreuzer und Stefan Kühling folgen. Die Kapitel des Buches lauten:

Was ist Logik?

Aussagenlogik

Hornlogik

Prädikatenlogik

Gleichungslogik

Modallogik

Temporallogik

Weitere logische Systeme

Hilbert-Kalküle

010117 **Übung zu Logik für Informatik**

Schwentick, Thomas

Vorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort

wird noch bekannt gegeben

040305 **Funktionale Programmierung**

Padawitz, Peter

Vorlesung 1 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Freitag	14:15	15:00	wöchentlich	OH-14 / E23

040306 **Übung zu Funktionale Programmierung**

N.N.

Übung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	12:15	14:00	wöchentlich	OH 16 / E 07
	Dienstag	10:15	12:00	wöchentlich	OH-14 / 104

Kurs

040505 Programmierkurs für Umsteiger (Wechsel von Diplom- in Bachelorstudiengänge)

Dißmann, Stefan

Praktikum 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
wird noch bekannt gegeben					

Kommentar

Der Programmierkurs für Umsteiger richtet sich speziell an solche Studierende, die von einem Diplom- in einen Bachelorstudiengang wechseln wollen und die Prüfung DAP 1 bereits bestanden haben. Der Kurs ermöglicht es diesen Studierenden, die ihnen noch für das Modul DAP 1 fehlende Studienleistung für Element Programmierpraktikum ohne erneuten Besuch der Vorlesung DAP 1 zu erwerben.

Der Programmierkurs bietet die notwendigen Inhalte und Leistungskontrollen als Kompaktveranstaltung in der vorlesungsfreien Zeit nach dem Wintersemester 2007/08 an. Um die Studienleistung nachzuweisen müssen vorgegebene Aufgaben gelöst werden, deren Problemstellungen sich an den Inhalten der Vorlesung DAP 1 orientieren.

Praktika

040501 Hardwarepraktikum für Informatiker

Jansen, Winfried ; Temme, Karl-Heinz

Praktikum 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	14:00	18:00	wöchentlich	OH 16 / U 12
	Mittwoch	14:00	18:00	wöchentlich	OH 16 / U 12

Voraussetzungen DPO 2001: Fachprüfung Rechnerstrukturen

Kommentar Anmeldungen via Internet ab September 2005

Literatur Skript erforderlich. Wann und wo erhältlich wird noch bekannt gegeben.

Bemerkungen DPO Informatik (5.6.1996) § 10 wird wie folgt geändert: In Absatz 5 erhält Nr. 1 folgenden Wortlaut: "1. a) für Studierende mit Nebenfach Elektrotechnik: ein Leistungsnachweis über die Teilnahme am Digitalelektronischen Praktikum für Informatik-Studierende. Zulassungsvoraussetzung für das Digitalelektronische Praktikum ist die bestandene Fachprüfung in Grundlagen für Elektrotechnik I und II. b) für Studierende, die nicht das Nebenfach Elektrotechnik gewählt haben: ein Leistungsnachweis über die Teilnahme am Hardware-Praktikum sowie als Zulassungsvoraussetzung für das Praktikum ein Leistungsnachweis über eine erfolgreiche Teilnahme an der Lehrveranstaltung "Grundlagen der Schaltungstechnik für Informatik-Studierende"."

Leistungsnachweis Ja.

040503 Programmierpraktikum DAP 1

Dißmann, Stefan ; Eskin, Esmeray ; Menge, Sebastian ; Rupflin, Wilfried ; Sugioarto, Martin

Praktikum 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
					wird noch bekannt gegeben

Kommentar Das Ziel des Programmierpraktikums ist der Erwerb von Programmiererfahrungen in der Programmiersprache JAVA. Dazu müssen vorgegebene Aufgaben gelöst werden, deren Problemstellungen sich an den Inhalten der Vorlesung DAP 1 orientieren.

040508 Softwarepraktikum in der vorlesungsfreien Zeit

Schmedding, Doris

Übung 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
					keine Angabe Kompaktphase /

Kommentar Ziel des Software-Praktikums ist die Vermittlung von praktischen Erfahrungen mit Methoden und Verfahren aus der Software-Technik. Dazu führen Gruppen von 7-9

Studierende Softwareentwicklungsprojekte durch. Ausgehend von einer groben Aufgabenstellung führen die Arbeitsgruppen im ersten Projekt mit Unterstützung ihres Betreuers alle Phasen des Software-Entwicklungsprozesses durch. In jeder Phase werden Dokumente erstellt, die von den Veranstaltern des Praktikums korrigiert werden. Das erste Projekt dient auch dem Zusammenwachsen der Arbeitsgruppen. Projekt 2, das den gleichen Ablauf wie Projekt 1 besitzt, dient dazu, das eingeübte Vorgehen zu vertiefen und zuvor gemachte Fehler zu vermeiden. Die eingesetzte Programmiersprache ist Java, die Modellierung erfolgt mit UML. Nähere Einzelheiten finden sich auf den Web-Seiten des Sopras.

Proseminare

040727 Meilensteine der Theoretischen Informatik

Bollig, Beate

Proseminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	08:15	10:00	wöchentlich	OH-14 / 304
Voraussetzungen	DAP 2, wünschenswert GTI				
Kommentar	In diesem Proseminar wollen wir uns mit herausragenden Ergebnissen der theoretischen Informatik vertraut machen, insbesondere mit solchen, die eine überraschende oder kreative neue Methode im Beweis verwenden. Die ausgesuchten Themen stammen aus den Bereichen Berechenbarkeit, Logik, Komplexitätstheorie, Schaltkreistheorie und Algorithmik. Beispielsweise wollen wir uns beschäftigen mit: - Untere Schranken für die Schaltkreiskomplexität der Paritätsfunktion - Äquivalenzprobleme und untere Schranken bei Branching-Programmen - PAC-Lernen und Occams Razor - - Probabilistische Algorithmen, Wahrscheinlichkeitsverstärkung und Recycling von Zufallszahlen - Interaktive Beweise und Zero Knowledge - $IP=PSPACE$ - Pebble Game				
Literatur	Uwe Schöning (1995). Perlen der Theoretischen Informatik. BI Wissenschaftsverlag.				
Bemerkungen	Die Vorbesprechung (inklusive Themenvergabe) findet am 4.10.2007, 14:15 Uhr, im Raum 3.05 OH14 statt. Interessenten können sich jedoch bereits vorab bei der Veranstalterin melden.				

040728 Algorithmen für NP- vollständige Probleme

Björklund, Henrik

Proseminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	12:15	14:00	wöchentlich	OH 16 / 205
Voraussetzungen	DAP 2				
Kommentar	Die NP-vollständigen Probleme sind eine große und wichtige Klasse von Problemen, für die wir keine effizienten (polynomiellen) Algorithmen kennen. In dieser Klasse finden sich unter anderem Probleme wie Erfüllbarkeit für aussagenlogische Formeln, das Travelling Salesman Problem, Linearprogrammierung mit ganzen Zahlen und das Rucksackproblem. Obwohl wir keine effizienten Algorithmen kennen, kommen diese Probleme in der Praxis immer wieder vor und müssen gelöst werden. In diesem Proseminar studieren wir die Methoden, die dazu verwendet werden.				

Mögliche Themen des Proseminars sind:

- Randomisierte Algorithmen
- Parametrisierte Analyse
- Local Search
- Approximationsalgorithmen

Literatur

Juraj Hromkovic: Algorithmics for Hard Problems. Zweite Ausgabe. Springer 2004.

Bemerkungen

Erste Vorbesprechung: Mittwoch, den 18. Juli, 10:15 Uhr, Otto-Hahn-Str. 16, Raum 205. (Hier werden die ersten 4-6 Themen verteilt.)

Zweite Vorbesprechung: Mittwoch, den 10. Oktober, 12:15 Uhr, Otto-Hahn-Str. 16, Raum 205. (Hier werden die restlichen Themen verteilt.)

Veranstaltungen im Hauptstudium

Pflichtveranstaltungen

041025 Übersetzerbau

Padawitz, Peter

Vorlesung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	14:15	16:00	wöchentlich	OH-14 / E23

Voraussetzungen Vordiplom

Kommentar Lehrinhalte: Die LV behandelt Grundlagen des Entwurfs und der Realisierung von Compilern: lexikalische Analyse (reguläre Sprachen, endliche Automaten, Scanner); konkrete und abstrakte Syntax; Syntaxanalyse (LL-, LR- und monadische Parser); semantische Analyse und Codeerzeugung (durch Attributierung und Transformation von Syntaxbäumen); vollständige Übersetzung einer prozeduralen, blockstrukturierten Sprache in Assemblercode; Typinferenz. Außerdem wird die funktionale Programmiersprache Haskell eingeführt, weil diese am geeignetsten ist, mit der Erzeugung, Attributierung und Auswertung von Syntaxbäumen befasste Algorithmen zu implementieren. Literatur: Skript zur LV (Kap. 1, 2.1, 2.2, 4.1-4.3, 5 und 6.1), <http://fdit-www.cs.uni-dortmund.de/~peter/Cbau.pdf> Appel, Modern Compiler Implementation in ML, Cambridge University Press 1998 Hopcroft, Motwani, Ullman, Einführung in die Automatentheorie, Formale Sprachen und Komplexität, Pearson Studium 2002 Kompetenzen: Es werden grundlegende Entwurfs- und Programmier Techniken für Algorithmen vermittelt, die mit der Erkennung, Transformation und Analyse symbolischer Daten zu tun haben. In diesem Kerngebiet der Informatik treffen sich Theorie und Praxis in besonders gut aufeinander abgestimmter Weise. Prinzipien, Handhabung und Anpassbarkeit der Techniken bilden die vor dinglichen Lernziele, weniger die Kenntnis eines bestimmten Werkzeugs (z.B. eines Compilergenerators), weil die nur in einem sehr begrenzten Umfeld von Nutzen wäre.

041026 Übung zu Übersetzerbau

Mattick, Volker ; Rüthing, Oliver

Übung

1 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	16:15	18:00	wöchentlich	OH 16 / E 07
	Dienstag	10:15	12:00	wöchentlich	OH-14 / 304
	Dienstag	16:15	18:00	wöchentlich	OH-14 / 104
	Mittwoch	16:15	18:00	wöchentlich	OH-14 / 104

041030 Unternehmensgründung (Informatik im Kontext)

Hack, Andreas; Witt, Peter

Vorlesung/Übung

3 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
------------	-----	-----	-----	----------	-----

Donnerstag 8:30 11:45 wöchentlich P1-05-306

2-tägige einführende Blockveranstaltung am 19.10.07 und am 26.10.07
danach Veranstaltungsteil Gründungsfinanzierung

Kommentar

Für mehr als 80 Millionen Euro kaufte im Januar 2007 der Holtzbrinck Konzern das junge Startup "StudiVZ". Nach nur eineinhalb Jahren Geschäftstätigkeit gelang den Gründern Ehssan Dariani (26), Dennis Bemann (28) und Michael Brehm (27) somit der Coup ihres Lebens.

Das Modul "Gründungsmanagement für Nicht-Wirtschaftswissenschaftler" richtet sich speziell an Studierende, die den Prozess der Unternehmensgründung verstehen und Anregungen für eine eigene Gründung erhalten wollen.

Dabei werden in einer einführenden zweitägigen Blockveranstaltung von der Ideenfindung über die Businessplanerstellung, der Ressourcenbeschaffung und den Markteinstieg, bis hin zum erfolgreichen Wachstum die vielfältigen Facetten einer erfolgreichen Gründung praxisnah beleuchtet.

Daran anschließend wird Semester begleitend in einer zweistündigen Veranstaltung gezeigt, wie Gründungsunternehmen ihren Kapitalbedarf bestimmen und sich entsprechend finanzieren können. Neben den Möglichkeiten der Fremdfinanzierung, z.B. durch Bankkredite und Förderkredite, liegt der Schwerpunkt der Veranstaltung auf den Möglichkeiten der Eigenfinanzierung von Gründungsunternehmen, z.B. durch Business Angel oder durch Venture Capital.

Ziel des Ausbildungsmoduls ist es, Nicht-Wirtschaftswissenschaftlern die betriebswirtschaftlichen Problemstellungen einer Unternehmensgründung näher zu bringen und potenzielle Gründer zur Umsetzung einer eigenen Geschäftsidee zu motivieren.

Bemerkungen

Aktuelle Informationen finden Sie unter der Veranstaltungsnummer 111902
<http://www.wiso.uni-dortmund.de/igm/de/content/V2/V21/V21.html>

041031 Projektmanagement aus der Praxis (Informatik im Kontext)

Uzquiano, José-Luis

Vorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	14:15	18:00	14-tägig	OH-14 / E23

Kommentar

Es werden in der Praxis erprobte und als Best-Practice anerkannte Methoden und Praktiken zum Projektmanagement von Softwareprojekten vermittelt. Diese Praktiken sind skalierbar und können für kleine Projekte mit wenigen Tagen Projektaufwand bis zu Großprojekten mit hunderten Projekttagen Aufwand angewendet werden. Die Methoden und Praktiken werden an hand konkreter Aufgaben eingeübt.

Die Vorlesung findet jeweils donnerstags in Blöcken zu 4 Stunden an folgenden Terminen statt:

25.10.07 Einführung und Rollen
15.11.07 Planung
29.11.07 Aufwandschätzung
13.12.07 Vorgehensmodelle und Qualitätssicherung
10.01.08 Risikomanagement und Teamprozesse
24.01.08 Kundenvortrag mit Fallstudie

Literatur

Führen; Leisten; Leben; Fredmund Malik, Campus Verlag; 2006

Managing Successful Projects with PRINCE2; Office of Government Commerce; 2005

The Mythical Men Month, Frederick P. Brooks, Jr.; Addison-Wesley Publishing Co.;

1995

Moderne Software-Architektur; Johannes Siedersleben; Dpunkt Verlag (Taschenbuch); 2004

Der Rational Unified Process: Eine Einführung; Philippe Kruchten; Addison-Wesley; 1999

Bemerkungen

Die Vorlesung findet im 14-tägigen Wechsel mit den Übungen statt.

Leistungsnachweis

Hinreichende Bearbeitung der Übungsaufgaben; Präsentation der Ergebnisse; Vorstellen eines realen Projekts inkl. Ausarbeitung.

041032 Übung zu Projektmanagement aus der Praxis (Informatik im Kontext)

Decker, Hans

Übung

1 SWS

Zeit & Ort

Tag

von

bis

Rhythmus

Ort

keine Angabe

14-tägig

/

Kommentar

s. Vorlesung

Bemerkungen

Wahlpflichtveranstaltungen

041121 Mensch-Maschine-Interaktion

Müller, Heinrich

Wahlpflichtvorlesung 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	12:15	14:00	wöchentlich	OH-14 / E23
	Donnerstag	12:15	14:00	wöchentlich	OH-14 / E23
Voraussetzungen	Als Voraussetzung für die Vorlesung genügen die im Grundstudium erworbenen Kenntnisse.				
Kommentar	Gegenstand der Vorlesung sind Techniken und Methoden der Informatik zur Realisierung und Gestaltung der Interaktion von Menschen mit rechnergestützten technischen Systemen. In der Vorlesung sollen methodische Grundkenntnisse zur Entwicklung von Systemen der Mensch-Maschine-Interaktion erlernt werden. Dabei soll sowohl dem Stand der Technik als auch grundsätzlichen Methoden im Hinblick auf zukünftige Entwicklung Rechnung getragen werden. Die Vorlesung wird Studierenden nach der DPO 2001 empfohlen, die als Spezialgebiet "Intelligente Systeme" wählen, insbesondere, wenn dort der Schwerpunkt auf dem Gebiet der graphischen Systeme liegen soll. Für Studierende nach älteren Diplomprüfungsordnungen ersetzt die Vorlesung die bisherige Stammvorlesung "Graphische Systeme". Als Themen der Vorlesung sind geplant: * Dreidimensionale interaktive Graphik: Displaytechnik, Farbtheorie, 3D-Geometrirepräsentation, Transformationen und Projektionen, Sichtbarkeitsberechnung, lokale Beleuchtungsmodelle, globale Beleuchtungsmodelle, Texturen, Level-of-Detail-Darstellung, Graphikprogrammierung * Eingabesensorik: Digitale Signalverarbeitung, ein- und zweidimensionale Digitalisierung, Abtasttheorem, Signalverbesserung, Mustererkennung, Bildverarbeitung und Computersehen * Mechanik, Kollision und Haptik * Sprachverarbeitung: Interpretation natürlicher Sprache, Verarbeitung gesprochener Sprache (Spracheingabe und Sprachausgabe) * Metaphern in der Mensch-Maschine-Interaktion: Textbasierte Interaktion, ortsbasierte Interaktion, zeichnungsorientierte Interaktion, Gesten, Sprechen und Hören, immersive virtuelle Umgebungen, Hypertext und Hypermedia, wissensbasierte Interaktionsunterstützung * Entwicklung interaktiver Systeme: Entwicklungsprozess, Modelle von Benutzungsschnittstellen und Spezifikationsmethoden, Entwurfprinzipien, Gestaltungsrichtlinien, Implementierung interaktiver Systeme, Evaluierung interaktiver Systeme * Physiologie und Psychologie: Sehsinn, Hörsinn, Tastsinn, Kognition				
Literatur	Es wird ein Folienskript herausgegeben. Ferner ist geplant, ergänzendes Lesematerial in elektronischer Form zur Verfügung zu stellen.				
Bemerkungen	In den Übungen sollen ausgewählte Konzepte, die in der Vorlesung theoretisch präsentiert wurden, praktisch erprobt werden, um damit auch aktiv vertraut zu werden.				

041122 Übung zu Mensch-Maschine-Interaktion

Bollweg, Peter

Übung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	10:15	12:00	wöchentlich	OH 16 / 205
	Montag	16:15	18:00	wöchentlich	OH 16 / 205
	Mittwoch	10:15	12:00	wöchentlich	OH 16 / E 07
	Mittwoch	14:15	16:00	wöchentlich	OH 16 / E 07
	Donnerstag	10:15	12:00	wöchentlich	GB V / 420
	Donnerstag	10:15	12:00	wöchentlich	OH-14 / 104
	Freitag	10:15	12:00	wöchentlich	OH 16 / 205

041125 Eingebettete Systeme

Marwedel, Peter

Wahlpflichtvorlesung

4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	10:15	12:00	wöchentlich	OH-14 / E23
	Mittwoch	10:15	12:00	wöchentlich	HG II / HS 6

Voraussetzungen Die Vorlesung ist für Studierende nach dem Vordiplom gedacht. Typischerweise sollte sie unmittelbar nach dem Vordiplom, also im 5. Semester gehört werden. Sie ist für Studierende der Informatik, der Ingenieurinformatik und der Informationstechnik wie auch für Lehramtsstudenten geeignet. Kenntnisse der Vorlesung "Rechnerstrukturen" oder einer äquivalenten Vorlesung werden vorausgesetzt. Eine vorherige Teilnahme am Hardwarepraktikum oder einem äquivalenten Praktikum, soweit nicht ohnehin verpflichtend, wird empfohlen.

Kommentar Eingebettete Systeme sind Systeme, bei denen eine Informationsverarbeitung in eine Umgebung eingebettet ist und bei denen die Informationsverarbeitung vielfach überhaupt nicht wahrgenommen wird. Beispiele hierfür sind informationsverarbeitende Systeme in Fahrzeugen (Autos, Flugzeugen, Eisenbahnen), in der Mobilkommunikation und in der Fertigungstechnik. Die Informationsverarbeitung in diesen Systemen hat eine Reihe von weitgehend ähnlichen Anforderungen zu erfüllen: dazu gehören insbesondere die Verlässlichkeit, die Berücksichtigung von Zeitschranken und die Effizienz der Realisierung. Gemäß einer Vielzahl von Vorhersagen werden eingebettete Systeme in der Zukunft einen Großteil der Anwendungen der Informationsverarbeitung ausmachen. In der Vorlesung werden die Grundzüge solcher Systeme vermittelt. Die Vorlesung behandelt insbesondere:

- Spezifikationsprachen für eingebettete Systeme
- Hardware eingebetteter Systeme
- Realzeit-Betriebssysteme und Middleware
- Abbildung von Anwendungen auf Ausführungs-Plattformen
- Evaluation und Validierung
- Optimierte Nutzung knapper Ressourcen
- Test

In den Übungen wird der Entwurf eingebetteter Systeme exemplarisch auf der Basis von hierarchischen Zustandsautomaten sowie von neuen (!) Lego-Mindstorm-Robotern erprobt.

Literatur Die Vorlesung richtet sich nach dem Buch "Eingebettete Systeme" von P. Marwedel (Springer, 2007, ca. 29 Euro). Ältere englischsprachige Fassungen können weiterhin benutzt werden. Weitere Hinweise zu dem Buch sowie begleitende Folien sind über

die Webseite <http://ls12-www.cs.uni-dortmund.de/~marwedel/kluwer-es-book> zu erhalten.

Bemerkungen

Beginn am Mittwoch, den 17.10.2007 (Nicht am 16.10. !)

Leistungsnachweis

Für Studierende nach den neuen Prüfungsordnungen:

Studierende können zu dieser Lehrveranstaltung einen Leistungsnachweis erwerben oder eine Fachprüfung ablegen.

Leistungsnachweise über den erfolgreichen Besuch von Vorlesung und Übungen werden erteilt, sofern die Studierenden die von den Veranstaltern festgelegten Kriterien für eine erfolgreiche Bearbeitung der Übungsaufgaben erfüllen.

Eine Fachprüfung kann in Form einer Klausur am Ende des Semesters abgelegt werden. Die Bearbeitung der Übungsaufgaben bereitet auf die Teilnahme an der Fachprüfung vor. Für die Klausur wird in Absprache mit den Studierenden ein Wiederholungstermin festgelegt werden.

Vorlesung und Übung werden insgesamt 9 Leistungspunkte zugeordnet.

Anrechenbarkeit für Studierende, die nach den "alten" DPOs studieren:

* Für ("Kern"-) Informatiker: Aufgrund eines Beschlusses des Prüfungsausschusses gilt die Vorlesung als eine Praktische Stammvorlesung im Sinne der "alten" Prüfungsordnung für Informatik.

Alternativ ist es möglich, "Prozessrechnertechnik" als Spezialvorlesung zum Gegenstand einer Prüfung "Informatik III" zu machen.

Ausgeschlossen ist jedoch, sowohl eine Prüfung unter dem Titel "Eingebettete Systeme" in "Informatik II" als auch eine Prüfung unter dem Titel "Prozessrechner-technik" in "Informatik III" abzulegen.

* Für Angewandte Informatiker: Es ist weiterhin möglich, sich über die Wahlpflichtveranstaltung "Prozessrechner-technik" prüfen zu lassen. Auch hier kann eine Vorbereitung mittels Besuchs der Vorlesung "Eingebettete Systeme" erfolgen.

041126 Übung zu Eingebettete Systeme

Falk, Heiko ; Lokuciejewski, Paul

Übung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	08:15	10:00	wöchentlich	OH-14 / 104
	Dienstag	14:15	16:00	wöchentlich	OH-14 / 104
	Dienstag	16:15	18:00	wöchentlich	OH 16 / E 07
	Mittwoch	12:15	14:00	wöchentlich	OH-14 / 104
	Freitag	10:15	12:00	wöchentlich	OH 16 / E 07

041223 Darstellung, Verarbeitung und Erwerb von Wissen

Kern-Isberner, Gabriele

Wahlpflichtvorlesung

4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	10:15	12:00	wöchentlich	OH-14 / E23
	Donnerstag	10:15	12:00	wöchentlich	OH-14 / E23

Kommentar Die konsequente und effektive Nutzung von Wissen durch maschinelle Systeme lässt sich als Prozess bzw. Zyklus auffassen, in dem die Wissensrepräsentation eine zentrale Rolle einnimmt. Sie bestimmt im Wesentlichen den formalen Rahmen sowohl für die Wissensverarbeitung als auch für den Erwerb von Wissen. Dabei steht ein breites Spektrum von Methoden und Techniken zur Verfügung. Die Vorlesung vermittelt einen Überblick über wichtige Problemstellungen, mit denen sich automatisierte Wissens- und Informationsverarbeitung auseinandersetzen muss, und gibt eine Einführung in gängige Methoden für Darstellung, Verarbeitung und Erwerb von Wissen. Es werden u.a. folgende Themengebiete behandelt: - Klassische und nichtklassische Inferenzsysteme - Knowledge Engineering und Ontologien - Beschreibungslogiken - Schlussfolgern mit Default-Regeln - Unsicheres und subjektives Wissen - Aktionen und Planen - Agenten- Maschinelles Lernen und Wissensentdeckung

041224 Übung zu Darstellung, Verarbeitung und Erwerb von Wissen

Lochner, Jan-Hendrik

Übung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	12:15	14:00	wöchentlich	GB IV / 113
	Montag	12:15	14:00	wöchentlich	Pav. 6 / 18
	Montag	14:15	16:00	wöchentlich	Pav. 6 / 18
	Dienstag	08:15	10:00	wöchentlich	GB V / 420
	Donnerstag	12:15	14:00	wöchentlich	GB V / 420
	Donnerstag	14:15	16:00	wöchentlich	GB V / 420

041221 Komplexitätstheorie und Effiziente Algorithmen

Wegener, Ingo ; Sauerhoff, Martin

Vorlesung 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	12:15	13:45	wöchentlich	OH-14 / E23
	Mittwoch	12:15	13:45	wöchentlich	OH-14 / E23

Kommentar Diese Vorlesung hat im Wintersemester den Schwerpunkt Komplexitätstheorie und im Sommersemester den Schwerpunkt Effiziente Algorithmen. Ein Besuch beider Vorlesungen ist sinnvoll und für eine Spezialisierung in diesen Bereichen sehr nützlich.

Wenn wir in der glücklichen Lage sind, einen effizienten Algorithmus für ein von uns zu lösendes Problem gefunden zu haben, ist eine nahe liegende Frage, ob z.B. dessen Laufzeit noch wesentlich verbessert werden kann. Kann man zeigen, dass kein Algorithmus für dieses Problem asymptotisch eine bessere Laufzeit hat? Für viele praktisch wichtige Probleme sind andererseits bisher überhaupt keine effizienten Algorithmen bekannt -- wie können wir nachweisen, dass solche Probleme inhärent schwierig sind? Die Vorlesung Komplexitätstheorie beschäftigt sich mit den aktuell vorhandenen Möglichkeiten, solche Fragen zu beantworten.

In der Vorlesung GTI wurde bereits die NP-Vollständigkeitstheorie als wichtiger Teilbereich der Komplexitätstheorie vorgestellt. Unter der Annahme, dass P ungleich NP ist, können wir mit diesem Werkzeug für viele in der Praxis auftretenden Probleme (bzw. deren Abstraktionen) nachweisen, dass sie nicht exakt in Polynomzeit lösbar sind. Um für Optimierungsprobleme auch die Existenz von pseudopolynomiellen Algorithmen oder guten Approximationsalgorithmen ausschließen zu

können, werden Erweiterungen der bekannten Reduktionskonzepte benötigt, die hier in der Vorlesung vorgestellt werden.

Einen neuen Ansatz zur Untersuchung der Komplexität von Problemen stellen interaktive Beweissysteme dar, bei denen ein als "Beweiser" agierender Spieler eine als "Verifiziererin" agierende Partnerin mit Hilfe eines randomisierten Kommunikationsprotokolls davon überzeugen soll, Eingaben für ein bestimmtes Problem mit hoher Wahrscheinlichkeit richtig in Lösungen bzw. Nichtlösungen zu klassifizieren. Die in der Vorlesung vorgestellten interaktiven Beweissysteme für das Graphisomorphieproblem liefern ein starkes Argument dafür, warum dieses bisher nicht effizient lösbare Problem vermutlich nicht NP-vollständig ist.

Interaktive Beweissysteme stecken auch hinter einer der revolutionärsten Erfindungen der theoretischen Informatik der letzten Jahre, der PCP-Theorie (PCP = probabilistically checkable proof). Diese liefert die Möglichkeit, bezüglich der Güteschranken viel genauere Nichtapproximierbarkeitsergebnisse zu zeigen, als dies mit den klassischen Reduktionsmethoden möglich war. In der Vorlesung werden wir das PCP-Theorem als zentrales Ergebnis dieser Theorie kennen lernen und dessen Anwendungen diskutieren.

In den Anwendungen spielen randomisierte Suchheuristiken wie evolutionäre Algorithmen oder Simulated Annealing eine immer wichtigere Rolle. Diese Algorithmen nutzen nicht die volle Information über die Eingabe. Die Komplexitätstheorie reagiert auf alle neuen algorithmischen Entwicklungen und wir stellen eine Theorie vor, die Probleme darauf untersucht, ob sie mit randomisierten Suchheuristiken effizient lösbar sind.

Für viele Berechnungsmodelle, z.B. Schaltkreise und eingeschränkte Turingmaschinen, kann man die Schwierigkeit von Problemen nachweisen, indem man die zur Verfügung stehende "Hardware" in Teile zerlegt und argumentiert, dass zur Lösung des Problems viel Kommunikation zwischen diesen Teilen erforderlich ist. Dies liefert die Motivation für die Untersuchung von Kommunikationsspielen, bei denen die Teilnehmer jeweils nur einen Teil der Eingabe besitzen und gemeinsam eine Funktion auf der Gesamteingabe berechnen sollen, indem sie möglichst wenig miteinander kommunizieren. Wir behandeln einige wichtige Beispielpunkte für dieses Modell und wenden die erhaltenen Ergebnisse zum Nachweis von unteren Schranken an.

Insgesamt ergibt sich ein Einblick in die moderne Komplexitätstheorie mit überraschend vielen konkreten Ergebnissen, die den Entwurf effizienter Algorithmen in die richtigen Bahnen lenken.

Literatur

Die Vorlesung wird sich im Wesentlichen an folgendem Buch orientieren:
Ingo Wegener, "Komplexitätstheorie -- Grenzen der Effizienz von Algorithmen". Springer, 2003.
Geplant ist die Behandlung der Kapitel 7-12 und 14-16.

Die in der Vorlesung verwendeten Folien werden an dieser Stelle online zur Verfügung gestellt.

Bemerkungen

Schwerpunkt Komplexitätstheorie

041222 Übung zu Komplexitätstheorie und Effiziente Algorithmen (KT+EA)

Marquardt, Marcel

Übung

2 SWS

Zeit & Ort

Tag	von	bis	Rhythmus	Ort
Mittwoch	16:15	18:00	wöchentlich	OH-14 / 304
Freitag	12:15	14:00	wöchentlich	OH 16 / 205

042381 Formale Methoden des Systementwurfs 2

Steffen, Bernhard

Spezialvorlesung

4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	12:15	14:00	wöchentlich	OH-14 / 104
	Freitag	10:15	12:00	wöchentlich	OH-14 / 304

Kommentar Formale Methoden des Systementwurfs 2:

Die Veranstaltung beginnt am 16.10.2007

Komplexe Hard- und Softwaresysteme durchdringen immer mehr Lebensbereiche und werden zunehmend auch in Anwendungen eingesetzt, die hohe Anforderungen an Sicherheit und Verfügbarkeit stellen.

Prominente Beispiele sind Flugzeug- und Bremssteuerungen, aber auch Anwendungen im Finanzbereich. Hier stößt klassisches Testen an seine Grenzen. Daher werden formale Methoden in diesen Gebieten jetzt auch in der industriellen Praxis rigoros eingesetzt: Während Airbus schon lange auf formale Methoden setzt, hat Boeing erst kürzlich den Einsatz formaler Methoden für obligatorisch erklärt.

Formale Methoden zielen darauf ab, mit semantisch fundierten Techniken Aussagen über das Verhalten von Systemen zu beweisen oder automatisch zu berechnen.

Außer zu Validierungs- und Verifikationszwecken werden sie auch bei der maschinellen Weiterverarbeitung von Systembeschreibungen eingesetzt, z.B. in optimierenden Übersetzern oder im Kontext des jetzt modernen "Model Driven Design".

Die Vorlesung behandelt einen panoramischen Rundblick auf die relevanten Methoden zu Semantikbeschreibung, Analyse, Verifikation und modellbasiertem Test. Neben den klassischen Methoden für sequentielle Programme adressiert sie insbesondere auch Methoden für die Analyse und Verifikation verteilter Systeme, u.a. das sogenannte Model-Checking und modellbasiertes Testen.

Die Vorlesung wird wie ihre Vorgängerin im Sommersemester 2007 konsequent Tool-basiert aufgezogen. Alle grundlegenden Technologien werden in den Übungen sowohl theoretisch als auf praktisch, mit Tool- Einsatz behandelt.

Die Teilnahme an der Vorgängervorlesung im Sommersemester 2007 ist KEINE Voraussetzung!

042382 Übung zu Formale Methoden des Systementwurfs 2

Kubczak, Christian

Übung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	wird noch bekannt gegeben				

Wahlveranstaltungen

042397 Mustererkennung

Fink, Gernot

Spezialvorlesung

4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	16:15	18:00	wöchentlich	OH 16 / 205
	Donnerstag	14:15	16:00	wöchentlich	OH 16 / 205

Kommentar

Mustererkennung gehört zu den Bemühungen der modernen Informationstechnik, Wahrnehmungsleistungen zu automatisieren, wie sie sonst von natürlichen Vorbildern bekannt sind. Prominente Anwendungsfelder sind das Erkennen von Schrift, das Verstehen gesprochener Sprache und die Interpretation von Bildern. In verschiedenen Bereichen wie der Ökologie, der Mikrobiologie und der Robotik werden Mustererkennungsverfahren zunehmend zur Analyse von Meßsequenzen eingesetzt.

In diesem Kontext wird in der Vorlesung die Klassifikation von Mustern detailliert untersucht. Klassifikation bedeutet dabei, dass ein Muster als Gesamtheit einem Begriff - d.h. einer Klasse - zugewiesen wird. Im Gegensatz zu verwandten Forschungsdisziplinen (z.B. Künstliche Intelligenz) liegt der behandelte Schwerpunkt auf wahrscheinlichkeitstheoretischen Familien von Klassifikatoren, wie z.B. dem Bayes-Klassifikator oder Mischverteilungsklassifikatoren, und sogenannten verteilungsfreien Klassifikatoren. Darüber hinaus werden aktuelle diskriminative Methoden (z.B. kernelbasierte Verfahren wie Support Vector Machines) vorgestellt.

Die Veranstaltung richtet sich an Studierende zu Beginn ihres Haupt- bzw. Masterstudiums und dient v.a. als Grundlage (Basismodul) für weitergehende Veranstaltungen im Forschungsbereich "Intelligente Systeme". Beispiele für darauf aufbauende Veranstaltungen sind u.a. die Vorlesungen/Module "Spracherkennung" oder "Computer Vision". Darüber hinaus stellen die vermittelten Kenntnisse wichtige Voraussetzungen für ein weites Anwendungsfeld von statistischen Analysen realer Daten dar (z.B. Robotik, Bildinterpretation, Bioinformatik, Chemoinformatik etc.).

Literatur

Niemann, H.: Klassifikation von Mustern, Springer-Verlag, Berlin, 1983

Fukunaga, K.: Introduction to Statistical Pattern Recognition, Academic Press, Boston, 2. Auflage, 1990

Hastie T. et al., The Elements of Statistical Learning, Springer, 2001

Duda, R. O. et al., Pattern Classification, Wiley, New York, 2. Auflage, 2001

042398 Übung zu Mustererkennung

Fink, Gernot

Übung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	wird noch bekannt gegeben				

042417 Praktische Optimierung

Rudolph, Günter

Vorlesung 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	14:15	16:00	wöchentlich	OH 16 / 205
	Mittwoch	10:15	12:00	wöchentlich	OH 16 / 205

Voraussetzungen Vordiplom oder Bachelor
(Veranstaltung im Hauptdiplom bzw. Master Informatik)

Kommentar Bei der Optimierung komplexer Systeme speziell in den Ingenieurwissenschaften stellt sich meist schnell heraus, dass die Reichweite analytischer und exakter Lösungsmethoden wegen idealisierender Voraussetzungen für die Praxis zu eingeschränkt ist. Die 'Praktische Optimierung' behandelt deshalb solche Lösungsansätze, die sich für praxisrelevante Problemklassen wie die nichtkonvexe Optimierung unter dem Black-Box-Szenario, die Optimierung bei Unsicherheit sowie zeitvarianter Probleme, die mehrkriterielle und schließlich die symbolische Optimierung bewährt haben. Methodisch kommen hier direkte deterministische Suchverfahren als auch etwa evolutionäre Algorithmen zum Einsatz. Besonderes Augenmerk gilt der Hybridisierung der Optimierungsverfahren mit statistischen Methoden: Bei zeitinvarianten Problemen werden Prognosemodelle, bei der Optimierung unter Unsicherheit statistische Testverfahren, zur Funktionsapproximation etwa Krigingverfahren oder Neuronale Netze benutzt. Weitere Themen berühren softwaretechnische Fragen zur Kopplung von Optimierungsverfahren und (kommerziellen) Simulatoren sowie die sinnvolle Nutzung paralleler Hardware. In den Übungen soll sich mit den Lösungsansätzen aktiv auseinandergesetzt werden, wobei existierende Schnittstellen zu Simulatoren softwaretechnisch bedient werden müssen.

Neben dem Erwerb von Einsicht in die Problematik und analytische Struktur der jeweiligen Problemklasse sollen die Studierenden methodisches Spezialwissen zur praktischen Lösung solcher Probleme erlangen. Sie sollen die praxisorientierten Lösungsansätze kennen und beherrschen sowie die Fähigkeit besitzen, selbständig praxisrelevante Probleme bearbeiten zu können. Schließlich sollen die Ergebnisse auch kritisch beurteilt werden können.

048418 Übung zu Praktische Optimierung

Rudolph, Günter

Übung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	12:15	14:00	wöchentlich	OH-14 / 304

042379 Randomisierte Algorithmen

Bollig, Beate

Spezialvorlesung 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	14:15	16:00	wöchentlich	OH-14 / 304
	Donnerstag	08:15	10:00	wöchentlich	OH-14 / 304

Voraussetzungen	Es werden keine tieferen Kenntnisse aus der Wahrscheinlichkeitstheorie vorausgesetzt. Die entsprechenden Kenntnisse werden in der Vorlesung anhand von Anwendungen aus der Informatik entwickelt.
Kommentar	Die Veranstaltung ist dem Schwerpunktgebiet 4 zugeordnet. Randomisierte Algorithmen sind eine in den Anwendungen nützliche Verallgemeinerung deterministischer Algorithmen, solange die Wahrscheinlichkeit unerwünschter Verhaltensweisen wie zu lange Rechenzeiten oder die Berechnung falscher Ergebnisse sehr gering ist. Sie zeichnen sich häufig durch ihre Einfachheit und ihre Effizienz bei der Lösung komplexer Aufgaben aus. In der Vorlesung behandeln wir den Entwurf und die Analyse randomisierter Algorithmen. Beispielsweise wollen wir die folgenden Entwurfsmethoden kennenlernen: - Überlisten des Gegners - Methode der Fingerabdrücke - Wahrscheinlichkeitsverstärkung durch Wiederholungen - Stichprobenmethode - Methode der häufigen Zeugen - Zufälliges Runden Die Kenntnis dieser Entwurfsmethoden ist hilfreich bei der gezielten Suche nach einem effizienten randomisierten Verfahren für ein zu untersuchendes Problem.
Literatur	Juraj Hromkovic (2004). Randomisierte Algorithmen. Teubner Verlag. Rajeev Motwani, Prabhakar Raghavan (1995). Randomized Algorithms. Cambridge University Press. Weitere Literatur kann bei Bedarf bei der Dozentin nachgefragt werden.

042380 Übung zu Randomisierte Algorithmen

Bollig, Beate

Übung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	10:15	12:00	wöchentlich	OH-14 / 304

042371 Petrinetze

Dittrich, Gisbert

Spezialvorlesung 3 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	08:15	10:00	wöchentlich	OH 16 / 205
	Donnerstag	08:15	10:00	wöchentlich	OH 16 / 205

Kommentar	Diese Vorlesung ist in folgenden Schwerpunktgebieten verwendbar: 2 Rechnerarchitektur, eingebettete Systeme und Simulation
-----------	--

3 Verteilte Systeme 7 Intelligente Systeme

Petrinetze stellen eine graphisch orientierte Darstellungsform zur Beschreibung vor allem konkurrierender Systeme dar. Insbesondere zur Beschreibung von dynamischen Abläufen, wie etwa gewisser Phänomene bei Betriebssystemen, Produktionsprozessen und im Workflowmanagement werden sie eingesetzt. Dabei gibt es eine ganze Reihe variierender Ansätze zur Beschreibung dynamischer Abläufe auf Petrinetzen.

Ziel der Veranstaltung ist es, nach der Entwicklung der Grundideen aus der Welt der Petrinetze der Reihe nach die dort verwendeten Systembeschreibungen mit aufsteigender Beschreibungskomplexität einzuführen.

Zunächst werden daher
Elementare Netze,
Bedingungs-Ereignis Systeme sowie
Stellen-Transitions-Systeme

behandelt und ein Einblick in dort verfügbare theoretische Ergebnisse gegeben. Danach werden die für die praktische Modellierung wichtigeren sog. High-Level-Netze sowie hierarchische Netze behandelt, sofern die Zeit reicht.

Da die Vorlesung für drei SWS konzipiert ist, findet der Dienstags-Termin nur in der ersten Hälfte des Semesters statt.

Literatur

Literatur (auszugsweise):

- [ReRo98a] Reisig, W. ; Rozenberg, G. (Eds.): Lectures on Petri Nets I: Basic Models, LNCS 1491, Springer Verlag, 1998
- [ReRo98b] Reisig, W. ; Rozenberg, G. (Eds.): Lectures on Petri Nets II: Applications, LNCS 1492, Springer Verlag, 1998
- [Reis 86] Reisig, W.: Petrinetze - Eine Einführung, 2. Auflage, Springer 1986
- [Brau 80] Brauer, W. (Edt.): Net Theory and Applications, LNCS Vol. 84, Springer Verlag 1980
- [BrRR 87a] Brauer, W. ; Reisig ; W.-Rozenberg, G. (Edts.): Petri Nets: Central Models and Their Properties, LNCS Vol. 254, Springer Verlag 1987
- [BrRR 87b] Brauer, W. ; Reisig, W. ; Rozenberg, G. (Edts.): Petri Nets: Applications and Relationships to Other Models of Concurrency, LNCS Vol. 255, Springer Verlag 1987
- [Jens 92] Jensen, K.: Coloured Petri Nets, Volume 1, EATCS, Springer 1992

042373 Desktop Video

Dittrich, Gisbert

Spezialvorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	08:15	10:00	wöchentlich	OH 16 / 205

Kommentar Diese Vorlesung ist im Schwerpunktgebiet "7 Intelligente Systeme" verwendbar.

Aufgrund der rasanten technischen Entwicklung nimmt die Verwendung von Videos bzw. Videoclips rapide zu; z.B. im privaten Bereich als Filme über Urlaub oder Familienereignisse, im öffentlichen Bereich v.a. in der Werbung, in der Unterhaltung und in der Ausbildung (Education --> Edutainment). Dabei spielt der Rechner bei der Erstellung von Videos/Videoclips zunehmend eine größere Rolle. In dieser Spezialvorlesung für Informatiker werde ich mich sowohl mit den zugehörigen Grundlagen, dafür hinreichender Geräteausstattung als auch konkreten Anwendungen und deren Verbreitungsmöglichkeiten in Form von Arbeitsproben zu diesem Themenbereich befassen. Dies führt zu folgender (vorläufigen) Gliederung:

- Grundlagen/Hintergrund
- Tools zur Erstellung von Audio, Video und (hilfsweise) Animationen
- Methodische Aspekte der Erstellung von Videos
- Erstellung von Arbeitsproben
- Einsatzmöglichkeiten von Videos.

Literatur

Literatur (auszugsweise)
 Dittrich, G.: Desktop Video SS 2007, FB Informatik, UniDo
http://mediasrv.cs.uni-dortmund.de/Lehre/SS2007/Desktop_Video_SS2007/index.html
 Apple Computer, Inc.: Final Cut Pro 5 User`s Manual, 2005
 Brenneis, Lisa: Final Cut Pro 5 for Macintosh X, Peachpit Press, 2006
 Jordan, Larry: Final Cut Pro 5 Hands-on Training, Peachpit Press, 2006
 Cullen, Sean et al.:
 Optimizing your Final Cut Pro System, Peachpit Press, 2006
 Plank, Uli: Apple DVD Studio Pro 3, Galileo Press Bonn, 2005
 Effelsberg, W. - Steinmetz, R.: Video Compression Techniques, Dpunkt.verlag Heidelberg, 1998
 Dennings, R.: Fire on the Wire, The IEEE 1394 High Performance Serial Bus,
<http://files.keyfax.com/mlancentral/firewire.pdf>

042375 **Übersetzerbau 2**

Padawitz, Peter

Spezialvorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	16:15	18:00	wöchentlich	OH-14 / 104

042377 **Logisch-algebraischer Systementwurf 1**

Padawitz, Peter

Spezialvorlesung 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	14:15	16:00	wöchentlich	OH 16 / E 07
	Donnerstag	14:15	16:00	wöchentlich	OH 16 / E 07

042383 **SOS und Views II**

Steffen, Bernhard

Spezialvorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	14:15	16:00	wöchentlich	OH-14 / 104

Kommentar

SOS und Views

Die Veranstaltung beginnt am 22.10.2007

Strukturierte operationelle Semantiken (SOS) sind ein mächtiges und elegantes Mittel, um Programmiersprachen semantisch zu spezifizieren.

In der Vorlesung soll dieses Mittel um verschiedene andere Techniken wie Model-checking, temporallogischer Synthese und abstrakter Interpretation angereichert werden, um Views zu erzeugen, d.h. spezielle Sichten auf System- und Programmstrukturen.

042385 Geometrisches Modellieren

Müller, Heinrich

Spezialvorlesung 3 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	08:15	10:00	wöchentlich	OH-14 / E23
	Mittwoch	10:15	11:00	wöchentlich	OH-14 / E23
Voraussetzungen	Spezielle Voraussetzungen werden nicht benötigt. Allerdings sollte eine gewisse Zuneigung zur Mathematik bestehen.				
Kommentar	Das geometrische Modellieren (engl. Computer Aided Geometric Design = CAGD) befasst sich mit der Generierung von Beschreibungen geometrischer Formen wie Kurven, Flächen oder Körper. Kurven und Flächen finden vielfältige Anwendungen im Maschinenbau (Produktdesign, Automobile, Flugzeuge, Schiffe), in der Robotik, Text- und Bildverarbeitung, in der Medizin und der bildenden Kunst und im Electronic Commerce. Gegenstand dieser Vorlesung sind zunächst die grundlegenden Modellierungstechniken wie die Interpolation mit Polynomen und Splines, Coons-Flächen, Bezier-Technik, B-Spline-Technik, Netze, Unterteilungsverfahren, fraktale Kurven und Flächen. Ferner wird in die Kurven- und Flächenanalyse eingeführt, wobei neben den Methoden der Differentialgeometrie die Möglichkeiten der Visualisierung durch Algorithmen und Geräte der graphischen Datenverarbeitung vorgestellt werden. Schließlich ist der Einsatz der Modellierungstechniken in heutigen CAD-Systemen ein Thema dieser Vorlesung, wozu effiziente Algorithmen für Operationen zwischen Kurven und Flächen angegeben werden.				
Literatur	Es gibt verschiedene, meist englischsprachige Lehrbücher zum Thema dieser Vorlesung. An deutschsprachigen Büchern wird empfohlen: "S. Abramowski, H. Müller, Geometrisches Modellieren, BI-Wissenschaftsverlag, 1992"; "J. Hoschek, D. Lasser, Grundlagen der geometrischen Datenverarbeitung, B.G., Teubner, Stuttgart, 1989, 2. Aufl." Die Vorlesung orientiert sich weitgehend am erstgenannten Buch. Dieses Buch ist inzwischen im Buchhandel vergriffen, es wird jedoch als Skript herausgegeben. Ferner wird es ein Folienskript geben.				
Bemerkungen	Die Vorlesung wird durch Übungen ergänzt. Dort sollen die in der Vorlesung präsentierten Konzepte praktisch erprobt werden. Hierfür soll das Modellersystem MAYA eingesetzt werden. In der Vorlesung wird hierzu mehr gesagt.				

042386 Übung zu Geometrisches Modellieren

Müller, Heinrich

Übung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	wird noch bekannt gegeben				

042387 Datenvisualisierung

Müller, Heinrich

Spezialvorlesung

3 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	11:15	12:00	wöchentlich	OH-14 / E23
	Donnerstag	08:15	10:00	wöchentlich	OH-14 / E23
Voraussetzungen	Voraussetzung für die Vorlesung sind Mathematikkenntnisse, etwa wie sie im Rahmen des Grundstudiums der Informatik erworben werden. Ferner ist die Teilnahme an der Wahlpflichtvorlesung "Mensch-Maschine-Interaktion" hilfreich, aber nicht notwendig.				
Kommentar	Die Datenvisualisierung macht sich die hohe Bandbreite des menschlichen Sehsystems zunutze, um komplexe Prozesse und Simulationen, die eine große Datenmenge erzeugen, interpretieren und steuern zu können. Solche Prozesse treten in Naturwissenschaft und Technik, den Sozialwissenschaften und in der Informatik auf. Beispiele sind Strömungssimulationen, bildgebende tomographische Verfahren etwa in der Medizin, Software-Strukturen und Prozesszustände und das Internet. In den letzten zwanzig Jahren wurden unter anderem in dem Teilgebiet "Wissenschaftliche Visualisierung" der Computergraphik und dem der statistischen Datenvisualisierung eine Vielzahl von Verfahren entwickelt, verbunden mit dem Bestreben, diese anhand von Datencharakteristika zu strukturieren. In dieser Vorlesung werden Visualisierungsverfahren für ein breites Anwendungsspektrum und damit verbundene Algorithmen zu ihrer Realisierung vorgestellt. Geplante Themengebiete sind u.a. Visualisierung hochdimensionaler Daten, Visualisierung von Prozessen, Algorithmen und Software, Volumenvisualisierung, Visualisierung von Vektor- und Tensorfeldern, nichtfotorealistische Visualisierung, Einsatz von Systemen der virtuellen und erweiterten Realität (Virtual Environments and Augmented Reality) für die Visualisierung.				
Literatur	Zu der Vorlesung wird es ein Folienskript geben, das in großen Teilen auf Originalliteratur und Tutorientexten von Konferenzen aufbaut. Ein Eindruck von dem Gebiet kann etwa aus dem Sammelband "G. Nielson, H. Hagen, H. Müller (eds.), Scientific Visualization: Overview, Methodologies, Techniques, IEEE CS Press, 1997" gewonnen werden.				
Bemerkungen	In den vorlesungsbegleitenden Übungen sollen die in der Vorlesung angebotenen Konzepte praktische Anwendung finden. Es ist geplant, hierfür das Visualisierungssystem VTK zu verwenden.				

042388 Übung zu Datenvisualisierung

Müller, Heinrich

Übung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	wird noch bekannt gegeben				

042391 Automatisches Zeichnen von Graphen

Mutzel, Petra

Spezialvorlesung

4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	12:15	14:00	wöchentlich	OH-14 / 304
	Dienstag	12:15	14:00	wöchentlich	OH-14 / 304
Voraussetzungen	Vordiplom, hilfreich ist die VO Effiziente Algorithmen und Komplexitätstheorie				
Kommentar	Das Gebiet des Automatischen Zeichnens von Graphen beschäftigt sich mit Design, Analyse, Implementierung und experimenteller Evaluierung von Algorithmen zum automatischen Layout von diskreten Strukturen in der Ebene (oder im dreidimensionalen Raum). Anwendungen beinhalten Datenbankvisualisierungen, Datenmodelle im Businessbereich, UML-Klassendiagramme im Software-Engineering, Netzwerke in der Bioinformatik, sowie Entwurf und Analyse von Netzen in der Elektrotechnik. Kriterien für eine leicht verständliche (oder auch ästhetisch schöne) Zeichnung sind nach wahrnehmungspsychologischen Studien unter anderen: wenige Überkreuzungen zwischen Kanten, möglichst große Winkel zwischen den zu einem Knoten adjazenten Kanten, oder wenige Knicke in Kanten. Neben Algorithmen zum Zeichnen von allgemeinen Graphen und Digraphen werden wir auch Zeichenmethoden für Spezialklassen von Graphen behandeln, wie etwa von Bäumen, von planaren Graphen oder von Graphen mit Maximalgrad vier. Alle diese Verfahren behandeln das Zeichnen auf einer 2-dimensionalen Ebene, gegen Ende der Vorlesung werden wir Ausblicke zum Zeichnen in 3 Dimensionen geben. Die Methoden hierbei sind vielfältig und beinhalten sowohl einfache Algorithmen und Datenstrukturen (wie z.B. scan-line) als auch komplexe Graphenalgorithmen (z.B. lineare Planaritätstests, Netzwerkflussverfahren) und Optimierungsverfahren (z.B. Schnittebenenverfahren). Vermittelte Fähigkeiten: Analyse und Modellierung von Problemen, selbständige Implementierung einiger Zeichenverfahren, Einblick in die Graphentheorie, Graphenalgorithmen, Optimierungsverfahren				
Literatur	Aktuelle Originalarbeiten, z.B. in Lecture Notes in Computer Science 4372, Springer-Verlag, Kaufmann, M. und Wagner, D. (Eds.), Graph Drawing, 14th International Symposium, GD 2006, Karlsruhe, 2007 oder auch Mutzel, P. und Juenger, M.: Graph Drawing Software, Springer-Verlag, Berlin 2003.				
Bemerkungen	Schwerpunktgebiete: 4 und 1				

042392 Übung zu Automatisches Zeichnen von Graphen

Mutzel, Petra

Übung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	16:15	18:00	wöchentlich	OH-14 / 304

042395 Computer Vision

Fink, Gernot

Spezialvorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	14:15	16:00	wöchentlich	OH 16 / 205

Kommentar

For the majority of living beings vision is the most important perception mechanism for orienting themselves in the environment. Therefore, there exists a multitude of attempts to recreate this capability in artificial systems. In contrast to image processing techniques found in industrial applications the aim of such advanced systems for machine vision is to obtain a task-oriented interpretation of a complex scene with as few restrictions as possible concerning the context and the recording conditions.

In this lecture advanced techniques of machine vision are covered which to some extent are inspired by cognitive processes known from human visual perception. First, important aspects of imaging processes are introduced with an emphasis on the perception of colors. Afterwards, methods for the extraction of image primitives (e.g. regions and edges) and for the calculation of feature representations (e.g. texture, depth, or motion) are presented. Finally, the lecture focuses on visual perception processes at the boundary between image processing and scene interpretation. Especially, appearance based object recognition techniques and methods for tracking objects in image sequences will be covered.

The accompanying tutorials will give students the opportunity to deepen their knowledge of the theoretical concepts presented in the lecture by working on relevant practical problems.

Literatur

Gonzalez, Rafael C.; Woods, Richard E.: Digital Image Processing, Prentice Hall, 2nd Ed., 2002.

Forsyth, David A.; Ponce, Jean: Computer Vision - A Modern Approach, Prentice Hall, 2003.

042396 Übung zu Computer Vision

Fink, Gernot

Übung 1 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	wird noch bekannt gegeben				

042397 Mustererkennung

Fink, Gernot

Spezialvorlesung 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	16:15	18:00	wöchentlich	OH 16 / 205
	Donnerstag	14:15	16:00	wöchentlich	OH 16 / 205

Kommentar

Mustererkennung gehört zu den Bemühungen der modernen Informationstechnik, Wahrnehmungsleistungen zu automatisieren, wie sie sonst von natürlichen Vorbildern bekannt sind. Prominente Anwendungsfelder sind das Erkennen von Schrift, das Verstehen gesprochener Sprache und die Interpretation von Bildern. In verschiedenen Bereichen wie der Ökologie, der Mikrobiologie und der Robotik werden

Mustererkennungsverfahren zunehmend zur Analyse von Meßsequenzen eingesetzt.

In diesem Kontext wird in der Vorlesung die Klassifikation von Mustern detailliert untersucht. Klassifikation bedeutet dabei, dass ein Muster als Gesamtheit einem Begriff - d.h. einer Klasse - zugewiesen wird. Im Gegensatz zu verwandten Forschungsdisziplinen (z.B. Künstliche Intelligenz) liegt der behandelte Schwerpunkt auf wahrscheinlichkeitstheoretischen Familien von Klassifikatoren, wie z.B. dem Bayes-Klassifikator oder Mischverteilungsklassifikatoren, und sogenannten verteilungsfreien Klassifikatoren. Darüber hinaus werden aktuelle diskriminative Methoden (z.B. kernelbasierte Verfahren wie Support Vector Machines) vorgestellt.

Die Veranstaltung richtet sich an Studierende zu Beginn ihres Haupt- bzw. Masterstudiums und dient v.a. als Grundlage (Basismodul) für weitergehende Veranstaltungen im Forschungsbereich "Intelligente Systeme". Beispiele für darauf aufbauende Veranstaltungen sind u.a. die Vorlesungen/Module "Spracherkennung" oder "Computer Vision". Darüber hinaus stellen die vermittelten Kenntnisse wichtige Voraussetzungen für ein weites Anwendungsfeld von statistischen Analysen realer Daten dar (z.B. Robotik, Bildinterpretation, Bioinformatik, Chemoinformatik etc.).

Literatur

Niemann, H.: Klassifikation von Mustern, Springer-Verlag, Berlin, 1983
 Fukunaga, K.: Introduction to Statistical Pattern Recognition, Academic Press, Boston, 2. Auflage, 1990
 Hastie T. et al., The Elements of Statistical Learning, Springer, 2001
 Duda, R. O. et al., Pattern Classification, Wiley, New York, 2. Auflage, 2001

042398 **Übung zu Mustererkennung**

Fink, Gernot

Übung

2 SWS

Zeit & Ort

Tag

von

bis

Rhythmus

Ort

wird noch bekannt gegeben

042399 **Workflows in Theorie und Praxis**

Rehof, Jakob

Spezialvorlesung

2 SWS

Zeit & Ort

Tag

von

bis

Rhythmus

Ort

Montag

10:15

12:00

wöchentlich

OH-14 / 104

042401 **Modellierung und Simulation diskreter und kontinuierlicher Systeme**

Buchholz, Peter

Spezialvorlesung

2 SWS

Zeit & Ort

Tag

von

bis

Rhythmus

Ort

Dienstag

12:15

14:00

wöchentlich

GB V / 420

Voraussetzungen

Besuch der Veranstaltung "Modellgestützte Analyse und Optimierung" oder äquivalente Kenntnisse.

Kommentar

Die Vorlesung "Modellierung und Simulation diskreter und kontinuierlicher Systeme" vertieft die Inhalte der Wahlpflichtvorlesung "Modellgestützte Analyse und Optimierung".

Es werden die folgenden Themengebiete behandelt:

- Wiederholung wesentlicher Aspekte der stochastischen Simulation
- Verfahren zur Rangbildung und Bestenauswahl
- Experimentieren mit Simulatoren
- Optimierung von Simulationsmodellen
- Kontinuierliche Simulation
- Hybride Simulation
- Parallele und verteilte Simulation
- Techniken zur Varianzreduktion

Literatur

Es existiert kein Lehrbuch, das den Inhalt der Vorlesung abbildet.

Einige der Techniken werden in

A. M. Law, W. D. Kelton: Simulation Modeling and Analyse. McGraw Hill 2000 erläutert.

Zum Experimentdesign und zur Optimierung von Simulationsmodellen werden außerdem Resultate aus den Lehrbüchern

R. H. Myer, D. C. Montgomery. Response Surface Methodology. Wiley 2002 und

T. J. Santner, B. J. Williams, W. I. Notz. The Design and Analysis of Computer Experiments. Springer 2003 verwendet.

Für die kontinuierliche Simulation wurden die Einführungskapitel der Bücher

F. E. Cellier. Continuous System Modeling. Springer 1991

und

F. E. Cellier, E. Kofman. Continuous System Simulation. Springer 2006 verwendet.

Darüber hinaus wird Originalliteratur genutzt. Insbesondere ein die Tutorien der Winter Simulationskonferenzen (siehe <http://www.informs-cs.org/wscpapers.html>) sind zu empfehlen.

042402 Übung zu Modellierung und Simulation diskreter und kontinuierlicher Systeme

Buchholz, Peter

Übung

1 SWS

Zeit & Ort

Tag

von

bis

Rhythmus

Ort

Dienstag

14:15

16:00

wöchentlich

GB IV / 318

Kommentar

Die Übungen finden 14tägig statt. Übungstermine werden zu Beginn der Vorlesung abgesprochen. Die Übung ist teilweise projektorientiert, d.h. es werden einzeln oder in kleineren Gruppen komplexere Problemstellungen bearbeitet. Details zur Übung werden in der Vorlesung bekannt gegeben.

042405 Betriebssystembau

Spinczyk, Olaf

Spezialvorlesung

2 SWS

Zeit & Ort

Tag

von

bis

Rhythmus

Ort

Dienstag

16:15

18:00

wöchentlich

OH-14 / E23

Kommentar

Ziel der Lehrveranstaltung ist die Vermittlung von konzeptionellen Grundlagen und wichtigen Techniken, die für den Bau eines Betriebssystems erforderlich sind. In der vorlesungsbegleitenden Übung werden diese Kenntnisse praktisch angewendet, indem ein einfaches PC Betriebssystem in kleinen Arbeitsgruppen von Grund auf neu entwickelt wird. Um dies zu bewerkstelligen, sind fundierte Kenntnisse über Aufbau und Funktionsweise der PC Hardware erforderlich, die ebenfalls in der Lehrveranstaltung vermittelt werden. Angesprochen werden zum Beispiel das Schutzkonzept der IA-32 Architektur, aktuelle PC-Bussysteme und moderne Multi-Prozessor Interrupt Systeme. Gleichzeitig werden Grundlagen aus dem Betriebssystembereich wie

Unterbrechungen, Synchronisation und Ablaufplanung, die aus früheren Veranstaltungen weitgehend bekannt sein sollten, wiederholt und vertieft.
<http://ls12-www.cs.uni-dortmund.de/~spinczyk/BSB07/>

042406 Übungen zu Betriebssystembau

N.N.

Übung

4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	10:15	12:00	wöchentlich	OH 16 / E 07
	Donnerstag	12:15	14:00	wöchentlich	OH 16 / E 07

042407 Quantenalgorithmen und Quantenkryptographie

Sieling, Detlef

Spezialvorlesung

4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	12:15	14:00	wöchentlich	OH 16 / E 07
	Donnerstag	14:15	16:00	wöchentlich	OH-14 / 304

Kommentar

Quantenrechner sind ein Berechnungsmodell, von dem vermutet wird, dass man damit manche algorithmischen Probleme effizienter als mit gewöhnlichen klassischen Rechnern lösen kann. Beispielsweise kennt man zurzeit für das Problem der Faktorisierung von ganzen Zahlen keinen effizienten Algorithmus für klassische Rechner, wohl aber einen effizienten Algorithmus für Quantenrechner. Allerdings können derzeit nur winzig kleine Quantenrechner praktisch realisiert werden; an der Realisierung von größeren Quantenrechnern wird intensiv geforscht.

Die Sicherheit der meisten bekannten kryptographischen Systeme beruht auf Annahmen, dass bestimmte mathematische Probleme keine effizienten Algorithmen haben. Aus solchen Annahmen kann man dann folgern, dass es auch für das Brechen des jeweiligen kryptographischen Systems keinen effizienten Algorithmus gibt, das System also sicher ist. Insbesondere die Sicherheit des RSA-Systems beruht auf der Annahme, dass es keinen effizienten Algorithmus für das Faktorisieren von ganzen Zahlen gibt. Ob diese Annahme stimmt, ist derzeit unbekannt. Aber selbst wenn das Faktorisierungsproblem keinen effizienten Algorithmus für gewöhnliche Rechner hat, könnte es irgendwann einmal Quantenrechner geben, mit denen das RSA-System gebrochen werden kann.

In der Quantenkryptographie versucht man dieses Problem durch die Konstruktion von beweisbar sicheren kryptographischen Protokollen zu vermeiden. Für den Beweis der Sicherheit derartiger Protokolle benötigt man keine Annahmen über die Nichtexistenz von effizienten Algorithmen mehr. Die einzige Annahme, die für den Beweis der Sicherheit noch benötigt wird, besteht darin, dass die derzeit bekannten Naturgesetze, insbesondere die Quantentheorie, zutreffend sind.

In der Vorlesung sollen zunächst die Grundlagen für die Beschreibung von Quantenalgorithmen erarbeitet werden. Danach sollen die bekannten Quantenalgorithmen vorgestellt werden, nämlich der bereits erwähnte Faktorisierungsalgorithmus und ein Algorithmus für ein bestimmtes Suchproblem. Anschließend wird das BB84-Protokoll aus der Quantenkryptographie zusammen mit einem Beweis seiner Sicherheit behandelt.

Literatur

M.A. Nielsen, I.L. Chuang, Quantum Computation and Quantum Information. Cambridge University Press 2000.

M. Sauerhoff, Skript zur Vorlesung Quantenrechner: Algorithmen und Komplexität, 2003. Erhältlich unter <http://ls2-www.cs.uni-dortmund.de/lehre/sommer2003/QR/>

042409 Compiler für Eingebettete Systeme

Falk, Heiko

Spezialvorlesung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	14:15	16:00	wöchentlich	OH-14 / 104
Voraussetzungen	Notwendige Voraussetzungen: - Vordiplom optionale Voraussetzungen: - Vorlesung "Eingebettete Systeme"				
Kommentar	Die Bedeutung Eingebetteter Systeme steigt von Jahr zu Jahr. Innerhalb Eingebetteter Systeme steigt der Software-Anteil, der auf Prozessoren ausgeführt wird, aufgrund geringerer Kosten und höherer Flexibilität ebenso kontinuierlich. Wegen der besonderen Einsatzgebiete Eingebetteter Systeme kommen hier hochgradig spezialisierte Prozessoren zum Einsatz, die applikationsspezifisch auf ihr jeweiliges Einsatzgebiet ausgerichtet sind. So werden z.B. oft Digitale Signalprozessoren, Netzwerk-Prozessoren oder Prozessoren mit Parallelverarbeitung auf Instruktionsebene eingesetzt. Diese hochgradig spezialisierten Prozessoren stellen hohe Anforderungen an einen Compiler, der Code von hoher Qualität z.B. hinsichtlich Laufzeit oder Energieverbrauch generieren soll. Die Vorlesung "Compiler für Eingebettete Systeme" soll - den Einsatz von Compilern in den Entwicklungsprozess Eingebetteter Systeme einordnen, - Struktur und Aufbau derartiger Compiler aufzeigen, - interne Zwischendarstellungen auf verschiedenen Abstraktionsniveaus einführen, und hauptsächlich - Problemstellungen und Optimierungstechniken in allen Phasen des Compilers erläutern. Das Schwergewicht der Veranstaltung liegt klar auf dem letzten obigen Punkt. Aufgrund der hohen an Compiler für Eingebettete Systeme gestellten Anforderungen sind effektive Optimierungen unerlässlich. Hier soll bspw. darauf eingegangen werden, - welche Arten von Optimierungen es auf Quellcode-Niveau gibt, - wie der eigentliche Übersetzungsvorgang von der Quellsprache nach Assembler vorgenommen wird, - wie hochgradig spezialisierte Befehlssätze in der Code-Generierung durch Optimierungstechniken ausgenutzt werden können, - welche Arten von Optimierungen auf Assembler-Niveau durchzuführen sind, - wie die Registerallokation vonstatten geht, und - wie Speicherhierarchien effizient ausgenutzt werden. Da Compiler für Eingebettete Systeme oft verschiedene Zielfunktionen optimieren sollen (z.B. durchschnittliche oder worst-case Laufzeit, Energieverbrauch, Code-Größe), soll der Einfluß von Optimierungstechniken auf diese verschiedenen Zielfunktionen deutlich gemacht werden.				
Literatur	- P. Marwedel, Eingebettete Systeme, Springer 2007. - S. S. Muchnick, Advanced Compiler Design and Implementation, Morgan Kaufmann 1997. - A. W. Appel, Modern Compiler Implementation in C, Cambridge University Press 1998. - H. Falk, Source Code Optimization Techniques for Data Flow Dominated Embedded Software, Kluwer Academic Publishers 2004. - Diverse Original-Artikel.				

042411 Reaktive Sicherheit

Flegel, Ulrich ; Meier, Michael

Spezialvorlesung 2 SWS

Zeit & Ort	Tag von bis Rhythmus Ort Donnerstag 14:15 16:00 wöchentlich HG I / HS 3
Voraussetzungen	Die Hörer sollen über die zum Vordiplom verlangten Kenntnisse der Informatik und der Mathematik verfügen. Vorkenntnisse über andere Teilgebiete der Sicherheit sind nützlich, aber nicht notwendig.
Kommentar	Mit der wachsenden Abhängigkeit unserer Gesellschaft von der Zuverlässigkeit informationstechnischer Systeme (IT) gewinnen Fragen der IT-Sicherheit an Bedeutung. Während bisher vorrangig präventive Maßnahmen und Mechanismen im Vordergrund standen, wird zunehmend deutlich, daß IT-Sicherheit nicht allein durch Prävention erreichbar ist. Vielmehr stellt Prävention einen Grundpfeiler dar, neben dem ergänzend die reaktiven Aspekte der IT-Sicherheit stehen. Die Vorlesung zeigt auf, wo das Präventionsparadigma zu kurz greift und motiviert ergänzende Maßnahmen für eine reaktive Sicherheit. Die Hörer werden für Verwundbarkeiten sensibilisiert und für deren Entstehung bei der Programmierung. Anschließend wird in die Erkennung vorhandener Verwundbarkeiten sowie von Schadsoftware und Angriffen eingeführt. Einschlägige ausgewählte Techniken werden erläutert und ausgewählte Werkzeuge beschrieben. Es wird für die Wechselwirkungen mit dem Datenschutz besonders sensibilisiert. Der Zyklus der Vorfallsbehandlung wird mit Augenmerk auf seine technischen Aspekte eingeführt. Geplante Themen in Stichworten: * Präventive Sicherheit und wo sie zu kurz greift * Verwundbarkeiten * Verwundbarkeitsanalyse * Malware * Intrusion Detection * Vorfallsbehandlung * IT-Forensik
Literatur	Die Literaturangaben zur Vorlesung werden auf der Web-Seite bekanntgegeben.
Leistungsnachweis	Diese Veranstaltung kann für folgende Schwerpunktgebiete gewählt werden (vorläufige Information vorbehaltlich Genehmigung): * 3. Verteilte Systeme * 5. Sicherheit und Verifikation Die Fachprüfungen (gemäß DPO § 29 (3)) finden als mündliche Prüfung von in der Regel 15 bis 30 Minuten (gemäß DPO § 8) zu noch festzulegenden Terminen statt. Verbindliche Anmeldungen erfolgen im Sekretariat (GB V, 421 und 423) nach dem folgenden Verfahren: * Vorlegen einer unterschriebenen Anmeldung zur Fachprüfung * Auswählen eines verfügbaren (und keine Lücken im Prüfungsplan ergebenden) Prüfungstermins (werden noch festgelegt) * Gewählten Prüfungstermin verbindlich eintragen lassen und unterschriebene Anmeldung abgeben (die wir dann an das Prüfungsamt weiterleiten)

042412 Übung zu Reaktive Sicherheit

Flegel, Ulrich ; Meier, Michael

Übung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
					wird noch bekannt gegeben
Bemerkungen	Die Übungsblätter werden auf der Web-Seite veröffentlicht.				

042413 Kommunikationskomplexität

Sauerhoff, Martin

Spezialvorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	10:15	11:45	wöchentlich	OH-14 / 304

Kommentar

Alice in Aachen und Bob in Böblingen verfügen jeweils über eine lokale Kopie einer 1 Terabyte umfassenden Datenbank. Wie können die beiden mit möglichst geringem Verbrauch an Netzbandbreite feststellen, ob ihre Datenbanken konsistent sind? Tatsächlich kann man relativ einfach zeigen, dass ein optimales deterministisches Protokoll zum Informationsaustausch durch die triviale Vorgehensweise gegeben ist, bei der einer der beiden Teilnehmer dem jeweils anderen seinen kompletten Datenbankinhalt übermittelt. Wenn wir allerdings randomisierte Protokolle zulassen, kann das Problem durch Übertragung von nur logarithmisch vielen Bits in der Datenbankgröße gelöst werden, im konkreten Beispiel mit ungefähr 140 Bytes (statt 1 TB) bei einer Fehlerwahrscheinlichkeit von 1 %.

Wir werden in der Vorlesung das formale Modell der Kommunikationsprotokolle behandeln, mit dem sich Probleme wie das obige genau beschreiben und analysieren lassen. Ein wesentliches Ziel in der Kommunikationskomplexitätstheorie ist es, für elementare Probleme untere Schranken für den Kommunikationsaufwand nachzuweisen, der notwendig ist, um diese in verteilter Form zu lösen. Entsprechende Beweistechniken werden wir uns in der Vorlesung erarbeiten. Diese Ergebnisse haben vielfältige Anwendungsmöglichkeiten beim Nachweis von unteren Schranken für anwendungsnahe Berechnungsmodelle, wie z.B. für die OBDD-Größe, Tradeoffs zwischen Fläche und Zeit von VLSI-Schaltkreisen, Rechenzeit von 1-Band-Turingmaschinen, Tiefe von monotonen Schaltkreisen usw.

Grundlage der Vorlesung ist das Buch von Kushilevitz und Nisan (siehe Begleitmaterial) und Originalarbeiten. Nach einem Einführungsteil werden wir folgende fortgeschrittene Themen behandeln:

- Protokolle mit beschränkter Rundenanzahl;
- Mehrspielerprotokolle;
- elementare Informationstheorie und Anwendungen für den Nachweis von unteren Schranken;
- Anwendungen für Branchingprogramme und Zeit-Speicherplatz-Tradeoffs;
- Quantenkommunikationskomplexität.

Der Einführungsteil wird sich zum Teil mit dem entsprechenden Abschnitt aus der Vorlesung "Komplexitätstheorie" in diesem Semester überschneiden, da wir kein Wissen von dort voraussetzen.

Literatur

E. Kushilevitz, N. Nisan. "Communication Complexity". Cambridge University Press, 2006.

Weitere Literatur auf der Vorlesungs-Webseite:
<http://ls2-www.cs.uni-dortmund.de/~sauerhof/kkt0708>.

042415 Medienengineering

Kalkbrenner, Gerrit

Spezialvorlesung 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	14:15	16:00	wöchentlich	HG I / HS 2
	Donnerstag	08:30	10:00	wöchentlich	HG I / HS 2

Kommentar**Themen**

Die 6-stündige Lehrveranstaltung (4V+2Ü) Medienengineering befasst sich mit Technologie und Anwendungen rund um Multimedia. Neben der Erörterung der Grundlagen zu Medien (Text, Bild, Audio, Video) werden die Standards zur Medienorchestrierung vorgestellt.

Neben der Vorstellung aktueller Werkzeuge wird auch das methodische Vorgehen bei der Entwicklung von multimedialen Anwendungen diskutiert. Erforderliche Standards für Teledienste und Telepräsenz werden diskutiert.

Voraussetzungen

Voraussetzung für die Teilnahme an der Lehrveranstaltung ist ein abgeschlossenes Bachelor- oder Grundstudium.

Durchführung

Die Veranstaltung findet regelmäßig an drei Terminen statt. In den Vorlesungen werden nach einer ausführlichen Einleitung einzelne Themen vertieft dargelegt. Regelmäßig ausgegebene Übungsaufgaben werden im Rahmen der Übung mit den Teilnehmern diskutiert.

Leistungserfassung

Für die erfolgreiche Teilnahme an der Veranstaltung sind 70 % der Übungsaufgaben sowie eine abschließende mündliche Prüfung erforderlich.

042416 Übung zu Medienengineering

Kalkbrenner, Gerrit

Übung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	08:15	10:00	wöchentlich	OH 16 / E 07

Seminare

044651 Executable Specification Languages

Padawitz, Peter

Seminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	16:15	18:00	wöchentlich	OH 16 / E 07

044652 Intelligente Anwendungen im Internet

Morik, Katharina

Seminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	10:15	12:00	wöchentlich	GB IV / 113

Kommentar

Das World Wide Web (WWW) hat zu einer Fülle neuer Anwendungen geführt. Allerdings führt die technische Verfügbarkeit zu einer unübersichtlichen, sich täglich verändernden Menge von Dokumenten. Dadurch sind neuere Suchmaschinen, die Personalisierung, Web Mining und das Clustering von Web-Seiten wichtige Themen geworden. Die Beobachtung der Benutzung des Internet (Web Usage Mining) ist kommerziell interessant, kann aber auch z.B. zur Verbesserung des Zugriffs durch Caching genutzt werden. Auch die Bildung von communities kann durch Web Mining unterstützt werden.

In dem Seminar sollen die folgenden Themen besprochen werden:

1. Neuere Suchmaschinen
2. Änderungsverfolgung
3. Personalisierung
4. Web Usage Mining
5. Web Mining
6. Clustering

In der Vorbesprechung am Dienstag, 9.10.2007, 10 Uhr im GB IV, R. 110 (Besprechungsraum des LS8) werden die Referate verteilt.

Literatur

Ad 1)

F. Menczer (2002) Complementing Search Engines with online web mining agents, Decision Support Systems 992

A. Mc Callum, K. Nigam, J. Rennie, K. Seymore (1999) Building domain-specific search engines with machine learning techniques, AAAI Spring Symposium on Intelligent Agents in Cyberspace

T. Joachims (2002) Optimizing Search Engines using clickthrough data, SIGKDD-02 Ad 2)

A. Jatowt, K.K. Bun, M. Ishizuka (2004) Change Summarization in Web Collections, Procs. of the 17th international conference on Innovations in Applied Artificial Intelligence, 653 - 662 Ad 3)

T. Joachims, T. Mitchell, D. Freitag, R. Armstrong (1995) WebWatcher -- machine learning and hypertext, 7. Fachgruppentreffen Maschinelles Lernen, 1995 und AAAI Spring Symposium on Information Gathering, 6712

G. Kappel, B. Pröll, W. Retschitzegger, W. Schwinger (2001) Customisation for Ubiquitous Web Applications ? A Comparison of Approaches, Procs. of the Int. Conf. on Information Integration and Web-based Applications and Services

M. Eirinaki, M. Vazirgiannis (2003) Web Mining for Personalization, ACM Transactions on Internet Technology, 1-27 Ad 4)

R. Cooley, B. Mobasher, J. Srivastava (1999) Data Preparation for Mining World Wide Web Browsing Patterns, Knowledge and Information Systems journal, 1, 5-32

- A. Büchner, M.D. Mulvenna (1998) Discovering Internet Marketing Intelligence through Online Analytical Web Usage Mining, SIGMOD record 27, 4
- L. Paganelli, F. Paterno (2002) Intelligent Analysis of User Interactions with Web Applications, Procs. of the 7th int. conference on Intelligent User Interfaces, 111-118
- F. Bonchi, F. Giannotti, C. Cozzi, G. Manco, M. Nanni, D. Pedreschi, m C. Renso, S. Ruggieri (2001) Web log data warehousing and mining for intelligent web caching, Data and Knowledge Engineering journal, 39, 165-189 Ad 5)
- R. Kosala, H. Blockeel (2000) Web Mining Research ? A Survey, SIGKDD, 2, 1-15
- K. Liu, K. Bhaduri, K. Das, P. Nguyen, H. Kargupta (2006) Client-side web mining for community formation in peer-to-peer environments, SIGKDD explorations, 8, 11-20,
- N. Agarwal, H. Liu, J. Zhang (2006) Blocking objectionable web content by leveraging multiple information sources, SIGKDD explorations, 8, 17-26 Ad 6)
- M. Charikar, C. Chekuri, T. Feder, R. Motwani (1997) Incremental clustering and dynamic information retrieval, 626--635
- B. Mobashar, R. Cooley, J. Srivastava (1999) Creating Adaptive Web Sites through Usage-Based Clustering of URLs, In IEEE Knowledge and Data Engineering Workshop
- M. Wurst, K. Morik, I. Mierswa (2006) Localized Alternative Cluster Ensembles for Collaborative Structuring, ECML-06, 485-496

044653 Maschinelles Lernen und Wissensentdeckung

Morik, Katharina

Oberseminar

3 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	10:15	13:00	wöchentlich	GB IV / 113
Kommentar	Arbeiten zu maschinellem Lernen und zur Wissensentdeckung in Datenbanken werden gemeinsam gelesen und diskutiert. Diplomanden und Doktoranden sowie eingeladene Wissenschaftler stellen ihre Arbeiten vor. Besondere Aufmerksamkeit wird Problemen gewidmet, die "noch nicht" gelöst sind und sich Standardlösungen entziehen				
Bemerkungen	Dieses offene, wissenschaftliche Arbeiten ist nur in einem ausgesuchten Teilnehmerkreis möglich				

044654 Document Engineering

Doberkat, Ernst-Erich

Seminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	wird noch bekannt gegeben				
Kommentar	Die Verarbeitung umfangreicher und vielfach strukturierter Dokumente zeigt viele der Probleme, die auch im Lebenszyklus von Software-Systemen auftauchen. Fragestellungen zur Modularisierung treten neben Aspekte der Versionierung und Konfiguration, Meta-Informationen müssen gefunden und verwaltet werden, Referenztritt neben Wertsemantik, und die Struktur von Dokumenten stellt sich als wesentliches Bestimmungsmerkmal heraus. Daneben treten Aspekte der Repräsentation von Dokumenten und das Problem der Heterogenität. Große Dokumente werden selten von einer Person konstruiert, so dass sich die Frage nach Gruppenarbeit und ihrer Unterstützung durch Werkzeuge und allgemeiner durch eine Entwicklungsumgebung stellt. Wir wollen in diesem Seminar das Thema Document Engineering systematisch anhand der vorhandenen Literatur aus der Sicht der Softwaretechnik aufarbeiten. Da-				

her sollten Seminarteilnehmer zumindest Grundkenntnisse der Software-Technologie aufweisen können, also mindestens die Veranstaltung Softwaretechnik im Grundstudium erfolgreich absolviert haben, besser noch die Veranstaltung Software-Konstruktion im Hauptstudium. Jeder Teilnehmer präsentiert ein Thema, die zur Verfügung gestellte Literatur dient als erste Orientierung; wir erwarten also, daß sich die Seminarteilnehmer über ihr Thema in der Literatur weiter informieren. Der Seminar-Schein wird ausgegeben aufgrund regelmäßiger Teilnahme an den Sitzungen des Seminars, eines erfolgreich gehaltenen Seminarvortrags und der schriftlichen Ausarbeitung des Themas.

Bemerkungen

Vorbesprechung: Donnerstag, 12. Juli 2007, 10.15 im Pavillon, Raum 18, Campus Süd.
Die Themenliste hängt ab 2. Juli 2007 am Schwarzen Brett des Lehrstuhls für Software-Technologie aus, bei Fragen wenden Sie sich bitte per eMail an eskin@udo.cs.uni-dortmund.de

044655 Algorithm Engineering

Gutwenger, Carsten ; Mutzel, Petra

Seminar 2 SWS

Zeit & Ort

Tag	von	bis	Rhythmus	Ort
Mittwoch				OH-14 / 202

Voraussetzungen

Diese Veranstaltung ist ein Seminar für Studierende im Hauptstudium.

Kommentar

Algorithm Engineering beinhaltet das Design von Algorithmen, ihre theoretische Analyse sowie ihre experimentelle Evaluierung und berücksichtigt insbesondere die Aspekte realer Rechnerarchitekturen.

Die klassische Algorithmik beschränkte sich lange Zeit auf die Theorie. Viele Publikationen beschreiben die Algorithmen und Datenstrukturen unter idealen Voraussetzungen, die jedoch in der Praxis häufig nicht gegeben sind. Auch wurden zunehmend hochkomplexe Datenstrukturen verwendet, die allgemein als praktisch nicht implementierbar galten. Mit der zunehmenden Anwendung der theoretischen Ergebnisse in der Praxis merkte man, dass eine große Lücke zwischen Theorie und Praxis existiert. Das Gebiet des Algorithm Engineering soll dem entgegenwirken, indem es Algorithmen für die Praxis entwirft und analysiert, evaluiert und darauf aufbauend verbessert.

Thema des Seminars sind aktuelle Forschungsarbeiten des Algorithm Engineering zum Design von Algorithmen für aktuelle Prozessorarchitekturen (Cache-Hierarchien, Multi-Core Prozessoren, Prefetching, Branch-Prediction) und zu Anwendungen wie Graphenzeichnen, Netzwerk-Design und Tourenplanung.

Schwerpunktgebiete:

- * Algorithmen, Komplexität und formale Modelle
- * Computational Intelligence and Natural Computing
- * Intelligente Systeme
- * Rechnerarchitektur, eingebettete Systeme und Simulation

Bemerkungen

Die Vorbesprechung und Themenvergabe fand bereits im Juli 2007 statt.

Termine siehe:

<http://ls11-www.cs.uni-dortmund.de/people/paul/SeminarAE.jsp>

044656 Relationales Lernen und Modellieren

Kern-Isberner, Gabriele

Seminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	10:15	12:00	wöchentlich	GB V / 420

Kommentar Aussagenlogik ist ein bewährtes Fundament, um allgemeine Zusammenhänge zwischen Merkmalen u.Ä. darzustellen. Daher stellen aussagenlogische Methoden nach wie vor ein wichtiges Instrumentarium sowohl für die Wissensrepräsentation als auch für das maschinelle Lernen bereit. Sie sind jedoch nicht geeignet, um Beziehungen zwischen Objekten bzw. Klassen von Objekten abzubilden. Für Einsatzbereiche wie das Semantic Web oder Social Network Analysis sind relationale Ansätze daher unverzichtbar.

Die Themen dieses Seminars werden sich mit relationalen Repräsentationsformen beschäftigen, die seit einigen Jahren insbesondere im Bereich des maschinellen Lernens verwendet und ausgebaut werden. Dazu gehören vor allem die Induktive Logische Programmierung und diverse probabilistische relationale Netzwerkansätze. Weiterhin sollen typische Anwendungsgebiete solcher Methoden behandelt werden.

044657 Eingebettete Systeme

Marwedel, Peter

Blockseminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	wird noch bekannt gegeben				

Voraussetzungen Kenntnisse über den Inhalt der Vorlesung "Eingebettete Systeme" werden vorausgesetzt.

Kommentar Ankündigung der Themen: bis 11.11.2007 auf der Webseite des Veranstalters

Themenvergabe: ab 19.11.2007 im Sekretariat des Lehrstuhls Informatik XII

Durchführung: 19.-21.2.2008

044658 Simulated Annealing

Jansen, Thomas

Seminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	12:15	14:00	wöchentlich	OH-14 / 304

Kommentar Simulated Annealing ist eine allgemeine randomisierte Suchheuristik, die häufig erfolgreich praktisch eingesetzt wird und theoretisch gut untersucht ist. Im Rahmen des Seminars werden verschiedene Aspekte von Simulated Annealing beleuchtet, dabei liegt der Schwerpunkt in der Analyse der Performanz bei der Anwendung im Optimierungskontext.

Ausgehend von ein oder zwei Arbeiten aus der unten stehenden Literaturliste sollen jeweils für alle Seminarteilnehmerinnen und Seminarteilnehmer nachvollziehbare

Vorträge die verschiedenen Aspekte vorstellen und zur Diskussion stellen. Zu jedem Vortrag ist außerdem eine schriftliche Ausarbeitung zu erstellen, in der die benutzte Literatur prägnant und verständlich zusammengefasst ist.

Literatur

- S. Droste, T. Jansen, I. Wegener (2001): Dynamic parameter control in simple evolutionary algorithms. In W.N. Martin, W.M. Spears (Hrsg.): Foundation of Genetic Algorithms 6 (FOGA 2000). Morgan Kaufmann, 275--294.
- L. Ingber (1993): Simulated Annealing: Practice versus theory. Mathematical and Computer Modelling 18(11):29--57.
- T. Jansen, I. Wegener (2007): A comparison of simulated annealing with a simple evolutionary algorithm on pseudo-boolean functions of unitation. Theoretical Computer Science. Erscheint.
- M. Jerrum, G.B. Sorkin (1998): The Metropolis algorithm for graph bisection. Discrete Applied Mathematics 82(1):155--175.
- S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi (1993): Optimization by simulated annealing. Science 220:671--680.
- K. Meer (2007): Simulated Annealing versus Metropolis for a TSP instance. Information Processing Letters. Erscheint.
- W. Michiels, E. Aarts, J. Korst (2007): Theoretical Aspects of Local Search. Springer. daraus: Kapitel 8: Asymptotic convergence of simulated annealing. 149--189.
- D. Mitra, F. Romeo, A. Sangiovanni-Vincenti (1986): Convergence and finite-time behavior of simulated annealing. Advances in Applied Probability. D. Mitra, F. Romeo, A. Sangiovanni-Vincenti (1986): Convergence and finite-time behavior of simulated annealing. Advances in Applied Probability 18(3):747--771.
- A. Nolte, R. Schrader (2000): A note on the finite time behavior of simulated annealing. Mathematics of Operations Research 25(3):476--484.
- G.B. Sorkin (1991): Efficient simulated annealing on fractal energy landscapes. Algorithmica 6(1):367--418.
- K. Steinhöfel, A.A. Albrecht, C.K. Wong (1998): Various cooling schedules for simulated annealing applied to the job shop problem. Proceedings of the Second International Workshop on Randomization and Approximation Techniques in Computer Science. LNCS 1518, Springer, 260--279.
- I. Wegener (2005): Simulated annealing beats Metropolis in combinatorial optimization. Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (ICALP). LNCS 3580, Springer, 589--601.

044659 Algorithmische Spieltheorie

Sauerhoff, Martin

Seminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	12:00	14:00	wöchentlich	OH 16 / 205

Kommentar

Das Internet stellt eine Plattform dar, auf der viele unterschiedliche Individuen und Organisationen gemäß ihren eigenen Interessen miteinander interagieren. Wenn nun verteilte Algorithmen von solchen eventuell egoistisch oder gar betrügerisch handelnden Parteien ausgeführt werden sollen, müssen wir offensichtlich neben den üblichen technischen auch sozio-ökonomische Rahmenbedingungen beachten. Die zwei wichtigsten Anwendungsfelder sind der Betrieb des Internets selbst sowie der Bereich E-Commerce. Das noch junge Gebiet der algorithmischen Spieltheorie liefert den geeigneten Rahmen, um dieses Szenario formal zu behandeln. Dabei werden klassische Modelle aus den Wirtschaftswissenschaften kombiniert mit neuen algorithmischen Ideen.

Ein großes, gut entwickeltes Teilgebiet ist der (algorithmische) Mechanismenentwurf, bei dem es darum geht, Spielregeln so zu wählen, dass es im Interesse der Teilnehmer ist, zur Erreichung eines globalen Zieles zusammenzuarbeiten.

Themen:

- * Einführung in die klassische Spieltheorie;
- * Modelle für Routing mit egoistischen Teilnehmern;

- * Verteilter, algorithmischer Mechanismenentwurf;
- * Klassische Auktionentheorie;
- * Profitmaximierung in Auktionen;
- * Kombinatorische Auktionen.

Die Vorbesprechung (inklusive Themenvergabe) findet am 4.10.2007, 10:15 Uhr, im Raum 305, OH 14, statt. Interessentinnen und Interessenten können mich aber auch gerne bereits vorab per Email (sauerhof (at) cs.uni-dortmund.de) oder persönlich (OH 14, Raum 332) kontaktieren. [Persönliche Erreichbarkeit und regelmäßige Emailbearbeitung wegen Krankheit vermutlich erst ab September.]

044660 Suchmaschinen

Sauerhoff, Martin

Seminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	12:00	14:00	wöchentlich	OH 16 / 205

Kommentar	Für das World Wide Web sind aufgrund der anarchistischen Form der Datenspeicherung sowie dessen Größe und Dynamik andere Techniken zur Wiedergewinnung von Informationen notwendig als für klassische Datenbanken. Eine entscheidende Rolle spielen dabei Suchmaschinen, deren Technologie sich die Teilnehmerinnen und Teilnehmer dieses Seminars anhand von Originalarbeiten erarbeiten sollen. Themen: * Aufbau von Suchmaschinen; * Theorie von Ranking-Algorithmen wie z.B. PageRank; * Steuerung von Crawlern; * Hashingverfahren für die Entdeckung von ähnlichen Webseiten; * Sampling von Webseiten. Die Vorbesprechung (inklusive Themenvergabe) findet am 4.10.2007, 12:15 Uhr, im Raum 305, OH 14, statt. Interessentinnen und Interessenten können mich aber auch gerne bereits vorab per Email (sauerhof (at) cs.uni-dortmund.de) oder persönlich (OH 14, Raum 332) kontaktieren. [Persönliche Erreichbarkeit und regelmäßige Emailbearbeitung wegen Krankheit vermutlich erst ab September.]
-----------	---

044661 Mobile Learning 2

Kalkbrenner, Gerrit

Seminar

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	12:15	14:00	wöchentlich	OH 16 / E 07

Kommentar	Bei diesem Seminar handelt es sich um die Fortführung des gleichnamigen Seminars Mobile Learning aus dem SoSe 07. Nachdem E-Learning eine umfangreiche Akzeptanz erfahren hat und Kleingeräte für den Mobilgebrauch (Handys, PDA, PPS, andere) verfügbar sind, rücken Lernarrangements für das mobile E-Learning (M-Learning) in den Fokus von Forschung und Entwicklung. Dieses Seminar befasst sich mit dem Potenzial und jüngsten Entwicklungen rund um M-Learning. Durchführung Die Veranstaltung findet regelmäßig an Dienstagen statt. Nach einer Einleitung des Dozenten werden weitere Themen von den Teilnehmern vorgetragen. Leistungserfassung Für die erfolgreiche Teilnahme an der Veranstaltung ist ein Referat zu erarbeiten, im Kreis des Seminars zu halten und eine schriftliche Ausarbeitung zu erstellen.
-----------	--

Veranstaltungen für Lehramtstudierende

041301 Einführung in die Didaktik der Informatik

Vahrenhold, Jan

Vorlesung

2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	10:15	12:00	wöchentlich	OH-14 / 304
Kommentar	Die Vorlesung "Einführung in die Didaktik der Informatik" befasst sich mit grundlegenden Themen der Fachdidaktik Informatik. Speziell werden das Selbstverständnis der Fachwissenschaft, die Positionierung gegenüber verwandten Fächern und die hieraus resultierenden Konsequenzen für das Schulfach Informatik besprochen. Fallstudien sollen unter anderem die Vermittlung von Unterrichtsgegenständen auch an jüngere Schülerinnen und Schüler sowie allgemeine und fortgeschrittene Modellbildungs- und Modellbeschreibungstechniken thematisieren. Die Veranstaltung wird inhaltlich durch die Vorlesung "Didaktik der Informatik II" fortgeführt, die im Sommersemester angeboten wird. In dieser Vorlesung werden unter anderem die Didaktik konkreter Unterrichtsgegenstände sowie Fallstudien zur Auswahl und Anwendung technischer Mittel behandelt.				
Literatur	Als ergänzende Literatur zu den bereit gestellten Vorlesungsunterlagen kann das folgende Lehrbuch dienen: Schubert, Sigrid, und Andreas Schwill (2004): Didaktik der Informatik, Spektrum Akademischer Verlag, Heidelberg. Weitere Literatur wird zu den einzelnen Kapiteln gesondert angegeben.				

041309 Grundlagen der Oberstufeninformatik

Vahrenhold, Jan ; Berk, Dennis

Proseminar

3 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	16:15	19:00	wöchentlich	OH-14 / 202
Voraussetzungen	Erfolgreiche Teilnahme an der Veranstaltung "Einführung in die Didaktik der Informatik"				
Kommentar	Im Rahmen dieser Veranstaltung werden die theoretischen Grundlagen der im Zentralabitur vorgesehenen Themen sowie Methoden zu deren Vermittlung behandelt. Ein Teil der Veranstaltung widmet sich der Entwicklung, Einordnung und Bewertung von Haus- und Klausuraufgaben. Die Veranstaltung besteht aus einem einführenden Kurs "Präsentationstechniken" zu Beginn des Semesters, einer Blockveranstaltung im ersten Drittel des Semesters sowie den in Blöcken stattfindenden Vorträgen in der zweiten Semesterhälfte. Die genauen Termine werden mit den Teilnehmern abgestimmt. Zu dieser Veranstaltung findet am Mittwoch, 11.07.2007, 17:00 Uhr, eine Vorbesprechung im Schullabor (U-08, OH-16) statt. Bei dieser Vorbesprechung werden die Themen vorgestellt und die Seminarplätze vergeben; es ist also notwendig, an dieser Vorbesprechung teilzunehmen.				

	Es ist nicht notwendig, sich für die Vorbesprechung anzumelden. Hinweis: Aus planungstechnischen Gründen müssen bei der Vergabe von Seminarplätzen solche Studierende bevorzugt behandelt werden, die noch einen Leistungsnachweis im Modul 11 (LPO 2003) erwerben müssen.
Literatur	Wird bei der Vorbesprechung bekannt gegeben.
Bemerkungen	Diese Veranstaltung ist durch den Prüfungsausschuss Informatik folgenden Modulen zugeordnet worden: * Modellversuch "Gestufte Studiengänge in der Lehrerbildung": Veranstaltung "Proseminar" des 5. Studienseesters mit 3.0 SWS * Studiengang Informatik (GyGes) gemäß LPO 2003: Modul 11 (Didaktik der Informatik); von den 3.0 zu besuchenden SWS der Veranstaltung werden 2(!) SWS angerechnet.
Leistungsnachweis	Aktive Teilnahme an Seminarsitzungen, Halten eines akzeptablen Vortrags, Anfertigung einer akzeptablen schriftlichen Ausarbeitung.

Besondere Veranstaltungen

049991 Kolloquium des Fachbereichs Informatik

Dekan

Kolloquium 4 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	16:15	18:00	wöchentlich	GB IV / HS 112
	Donnerstag	16:15	18:00	wöchentlich	HG I / HS 2

049990 CI-Kolloquium

Jägersküpper, Jens

Kolloquium 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	16:15	18:00	wöchentlich	OH-14 / 304

Veranstaltungen der Ruhr-Universität Bochum für Studierende mit Nebenfach 'Theoretische Medizin'

Lehrveranstaltungen vor dem Vordiplom

209500 **Anatomie I**

Haarmann (RUB), ; Hegelmeier (RUB), ; Theiß (RUB),

Vorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Mittwoch	15:00	17:00	wöchentlich	M / E 29

209501 **Biochemie I**

Geyer (RUB), ; Goody (RUB),

Vorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Dienstag	08:15	10:00	wöchentlich	MPI / Hörsaal MPI

209502 **Physiologie I**

Hohnsbein (RUB), Joachim ; Luttmann (RUB),

Vorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Montag	12:15	14:00	wöchentlich	M / E 28

Lehrveranstaltungen nach dem Vordiplom

206003 **Hygiene**

Wilhelm (RUB),

Vorlesung 2 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	14:15	16:00	wöchentlich	Bo Med. / HMA 30 (Bochum)

Bemerkungen ab 22.11.2005

209503 Pathologie

Guzmany Rotaeché (RUB),

Vorlesung 3 SWS

Zeit & Ort	Tag	von	bis	Rhythmus	Ort
	Donnerstag	11:30	13:45	wöchentlich	/
Kommentar	Institut für Pathologie RUB, Seminarraum Bergmannsheilkliniken				
Bemerkungen	Seminarräume, Institut für Pathologie der Ruhr-Universität an den BG Kliniken Bergmannsheil				

Anhang

Lehrangebot zu den Bachelor- und Masterstudiengängen *Informatik* und *Angewandte Informatik*

(Stand: 17.9.2007)

Ab dem Wintersemester 2007/2008 bietet der Fachbereich Informatik die Bachelor- und Masterstudiengänge *Informatik* und *Angewandte Informatik* an.

Während für Studienanfänger in den Bachelorstudiengängen im Wintersemester keine besonderen Regelungen gelten, gibt es für Studierende, die aus den Diplomstudiengängen in die Bachelorstudiengänge wechseln sowie für Studierende in den Masterstudiengängen einige Übergangsregelungen.

Für Studierende im höheren Bachelor-Fachsemestern:

- Im WS 07/08 wird bereits erstmals das *Modul Logik und Funktionale Programmierung* angeboten. Umsteiger aus dem Diplom, die den Leistungsnachweis *Logik* schon erworben haben, können hier auch nur die Vorlesung und die Übung *Funktionale Programmierung* belegen.
- Es werden im WS 07/08 wahrscheinlich noch keine *Proseminare* im Umfang von 4LP angeboten, aber voraussichtlich im Sommersemester 08.
- Für Umsteiger aus dem Diplom wird in der vorlesungsfreien Zeit 13.2.-26.2.2008) das Programmierpraktikum zu *DAP 1* als Blockveranstaltung angeboten.
- *Fachprojekte* werden erstmals im Sommersemester 08 angeboten.

Für Studierende im ersten Master-Semester:

Es werden im Wintersemester 07/08 die folgenden Basismodule angeboten:

- *Komplexitätstheorie* (im Vorlesungsverzeichnis: *Komplexitätstheorie und Effiziente Algorithmen*)
- *Mustererkennung*
- *Praktische Optimierung*
- *Formale Methoden des Systementwurfs 2* (wird als Basismodul im Forschungsbereich *Software, Sicherheit und*
- *Verifikation* anerkannt

Im Sommersemester 08 werden weitere Basismodule angeboten.

Wichtiger Hinweis:

Da die meisten der oben genannten Veranstaltungen hauptsächlich von Studierenden der Diplomstudiengänge besucht werden, ist es wichtig, dass Sie bereits zu Beginn der Veranstaltung die Lehrenden darüber informieren, falls Sie Studentin oder Student der Bachelor- oder Masterstudiengänge sind, und die genauen Prüfungsmodalitäten mit ihnen klären.

Informationen zum Wechsel von den Diplomstudiengängen (nach DPO 03) in die Bachelorstudiengänge

(Stand: 17.9.2007)

Grundsätzlich ist ein Wechsel von den Diplomstudiengängen in die Bachelor-Studiengänge möglich, in die Masterstudiengänge aber nicht, es sei denn es liegt bereits ein Bachelor- oder Diplomabschluss vor.

Formal erfolgt der Umstieg durch einen Antrag auf Studiengangswechsel beim Zentrum für Studienangelegenheiten. Wenn Sie in ein höheres Fachsemester eingestuft werden möchten, benötigen eine Bestätigung des Prüfungsausschusses auf dem Antragsformular für den Studiengangswechsel. Wenden Sie sich dazu bitte an den stellvertretenden Vorsitzenden des Prüfungsausschusses, Prof. Fink. *Nach dem Wechsel des Studiengangs* ist es möglich, sich durch den Prüfungsausschuss bereits erbrachte Studienleistungen anerkennen zu lassen.

Für die Anerkennung von Studienleistungen gilt im Prinzip: Leistungen für Veranstaltungen, die in den Bachelor-Studiengängen gleichartig vorkommen werden anerkannt. Das gilt auch wenn im Bachelor eine Note vorgesehen ist, im Diplom aber nicht (wie beim Proseminar). In solchen Fällen wird ergibt sich die Gesamtnote für die Abschlussprüfung durch das gewichtete Mittel aller benoteten Module.

Anerkannt werden insbesondere:

- die Elemente Übung und Vorlesung für *BSRvS 1 und 2, Rechnerstrukturen, Informationssysteme, GTI*
- die Elemente Übung und Vorlesung *Rechnerstrukturen* (wird im Bachelor im Umfang von 9LP anerkannt!)
- die Elemente Übung und Vorlesung für *DAP 1 und DAP 2* (beide Programmierpraktika sind aber nachzuholen)
- die Elemente Übung und Vorlesung der einschlägigen Mathematik-Veranstaltungen
- die Elemente Übung und Vorlesung *Logik* für das Modul *Logik und funktionale Programmierung* (die Elemente Übung und Vorlesung *Funktionale Programmierung* müssen aber nachgeholt werden)
- die Leistungsnachweise *Softwaretechnik* und *Softwarepraktikum* für das Modul *Softwareentwicklung*
- die Elemente Übung und Vorlesung von Wahlpflicht-Veranstaltungen
- das *Proseminar* (wird im Bachelor im Umfang von 4LP anerkannt!)
- das *Hardware-Praktikum*
- eine erfolgreich absolvierte *Projektgruppe* im Sinne der DPO 01 für das Modul *Fachprojekt*.

Bereits absolvierte Fachprüfungen in Diplomstudiengängen können im Bachelor nicht (zur Notenverbesserung) wiederholt werden. Fehlversuche aus den Diplomstudiengängen bleiben bestehen.

Die Teilnahme an betreuungsintensiven Veranstaltungen, die nur in den Bachelorstudiengängen vorgesehen sind (z.B.: Programmierpraktikum in *DAP 1* und *DAP 2*, Fachprojekt), ist erst nach dem Wechsel in die Bachelor-Studiengänge möglich.

Im Wintersemester 07/08 wird jedoch das Praktikum zu *DAP 1* zusätzlich in der vorlesungsfreien Zeit 13.2.-26.2.2008) angeboten. Daran können auch Studierende teilnehmen, die den Wechsel noch nicht formal vollzogen haben. Es ist jedoch eine formlose Anmeldung durch eine e-Mail an stefan.dissmann@udo.edu bis zum 17.1.08 erforderlich.

Beim Wechsel von DPO'96 KI bzw. DPO'97 AI in den Bachelor werden sinngemäß die Äquivalenzregelungen für den Wechsel nach DPO'01 angewandt.

Beispielhafter zeitlicher Verlauf eines Informatik-Studiums nach DPO 01

Die Skizze soll exemplarisch einen möglichen Verlauf eines Studiums im Studiengang 'Informatik' mit dem (häufig gewählten) Nebenfach Betriebswirtschaftslehre (BWL) nach der Diplomprüfungsordnung in der Fassung vom 2.2.2004 zeigen.

Die Skizze geht von ‚normalen‘ Studienverhältnissen aus:

- Ein Vollzeitstudierender bringt pro Woche 50h seiner Zeit für direkt studienbezogene fachliche Tätigkeiten (Besuch von Lehrveranstaltungen, Hausarbeiten) auf.
- Der Studierende widmet sich seinem Studium motiviert, engagiert und konzentriert. Er besitzt Kenntnisse entsprechend der allgemeinen Hochschulreife.
- Die vorlesungsfreie Zeit wird mindestens zur Hälfte konzentriert für das Studium genutzt (Praktika, Programmierkurse, Nachbereitung und selbständige Vertiefung, Prüfungsvorbereitung).

Grundstudium (78h)

1. Semester

<i>Datenstrukturen, Algorithmen und Programmierung 1 (DAP1)</i>	4V 2Ü
<i>Rechnerstrukturen (RS)</i>	3V 2Ü
<i>Mathematik 1: Lineare Algebra und Analysis</i>	4V 2Ü
<i>BWL: Technik des betrieblichen Rechnungswesens</i>	2V
	13V 6Ü 19h

Die zeitliche Belastung von 19 SWS lässt Raum zur Überwindung von Anfangsschwierigkeiten.

Zur Selbstkontrolle werden korrigierte Übungsaufgaben zu *DAP1*, *RS* und *Mathematik 1* bearbeitet.

In *Mathematik 1* wird ein Leistungsnachweis (Schein) erworben.

In der vorlesungsfreien Zeit werden schriftliche Prüfungen (Klausuren) abgelegt. Sie finden zu Beginn der vorlesungsfreien Zeit statt und können, falls nicht bestanden, gegen Ende der vorlesungsfreien Zeit wiederholt werden:

- Klausur zu *DAP1*
- Klausur zu *RS*
- Klausur zu *Technik des betrieblichen Rechnungswesens*

Damit ist das erste Semester hoffentlich erfolgreich abgeschlossen.

Bei Startproblemen könnte die erste Veranstaltung des Nebenfachs BWL (*Technik des betrieblichen Rechnungswesens*) auch erst im 2. Semester gehört werden.

2. Semester

<i>Datenstrukturen, Algorithmen und Programmierung 2 (DAP2)</i>	4V 2Ü
<i>Elektrotechnik und Nachrichtentechnik</i>	2V 1Ü
<i>Betriebssysteme, Rechnernetze und verteilte Systeme 1 (BSRVS1)</i>	2V 1Ü
<i>Mathematik 2: Diskrete Strukturen und Algebra</i>	4V 2Ü
<i>BWL: Kostenrechnung und Controlling</i>	1V 1Ü
<i>BWL: Bilanzierung und Controlling</i>	1V 1Ü
	14V 8Ü 22h

Die zeitliche Belastung von 22h erhöht sich um weitere 2 Stunden, falls die erste Veranstaltung des Nebenfachs erst im zweiten Semester gehört wird. Auch das ist noch gut machbar, insbesondere wenn man berücksichtigt, dass 8 Übungsstunden enthalten sind.

Zur Selbstkontrolle werden korrigierte Übungsaufgaben zu *DAP2*, *BSRVS1* und *Mathematik 2* bearbeitet.

Leistungsnachweise (Scheine) werden in folgenden Fächern erworben:

- *Elektrotechnik und Nachrichtentechnik*,
- *BSRVS1*.

In der vorlesungsfreien Zeit werden folgende Prüfungen abgelegt:

- *DAP2*,
- *Mathematik 2*,
- *BWL: Kostenrechnung und Controlling*,
- *BWL: Bilanzierung und Controlling*.

Mit dem Ende des zweiten Semesters sollten die Anfangsschwierigkeiten überwunden sein, und die vorlesungsfreie Zeit kann genutzt werden, um vier Fachprüfungen erfolgreich abzulegen.

3. Semester

<i>Softwaretechnik</i>	2V 1Ü
<i>Hardware-Praktikum</i>	4P
<i>Betriebssysteme, Rechnernetze und verteilte Systeme 2 (BSRVS2)</i>	2V 1Ü
<i>Logik</i>	2V 1Ü
<i>Wahrscheinlichkeitsrechnung und mathematische Statistik</i>	2V 1Ü
<i>BWL: Einführung in die Wirtschaftsinformatik</i>	3V
	11V 4Ü 4P 19h

Das dritte Semester hat eine etwas verminderte Belastung von 19h und bietet somit Raum zum Wiederholen und Nacharbeiten, falls sich in den ersten beiden Semestern Lücken ergaben. Weiterhin besteht die Möglichkeit, das Proseminar aus dem 4. Semester vorzuziehen.

Zur Selbstkontrolle werden korrigierte Übungsaufgaben zu *Softwaretechnik*, *BSRVS2*, *Logik* sowie *Wahrscheinlichkeitsrechnung und mathematische Statistik* bearbeitet.

Leistungsnachweise (Scheine) werden in folgenden Fächern erworben:

- *Softwaretechnik*,
- *Hardware-Praktikum*,
- *Logik*,
- *Wahrscheinlichkeitsrechnung und mathematische Statistik*.

In der vorlesungsfreien Zeit werden folgende Prüfungen abgelegt:

- *BSRVS2*,
- *BWL: Einführung in die Wirtschaftsinformatik*.

In der vorlesungsfreien Zeit nach dem dritten Semester stehen nur zwei Prüfungen an. Der entstehende Freiraum wird genutzt, um das *Software-Praktikum* aus dem vierten Semester schon in der vorlesungsfreien Zeit zu absolvieren.

4. Semester

<i>Software-Praktikum</i>	4P
<i>Informationssysteme (IS)</i>	2V 1Ü
<i>Grundbegriffe der theoretischen Informatik (GTI)</i>	4V 2Ü
<i>Proseminar</i>	2S
<i>BWL: Theorie der Produktionswirtschaft</i>	2V 1Ü
	8V 4Ü 4P 2S 18h

Das vierte Semester hat wieder eine etwas verminderte Belastung von diesmal sogar nur 18h. Der Freiraum sollte genutzt werden, um unter Umständen „noch nicht geschaffte“ Prüfungen vorzubereiten und zu wiederholen. Ziel sollte sein, dass das Vordiplom mit dem Ende der vorlesungsfreien Zeit nach dem vierten Semester erfolgreich bestanden ist.

Zur Selbstkontrolle werden korrigierte Übungsaufgaben zu *IS* und *GTI* bearbeitet.

Leistungsnachweise (Scheine) werden in folgenden Fächern erworben:

- *Software-Praktikum*,
- *Proseminar*.

In der vorlesungsfreien Zeit werden folgende Prüfungen abgelegt:

- *IS*,
- *GTI*,
- *BWL: Theorie der Produktionswirtschaft*.

Das *Proseminar* sollte nicht als „notwendiges Übel“ mit minimalen Aufwand bearbeitet werden. Es lohnt sich, Zeit und Engagement einzubringen, weil hier unter direkter Betreuung Vorgehensweisen und Tätigkeiten geübt werden, die im Verlauf des späteren Hauptstudiums beherrscht werden müssen (Literaturrecherche, Einarbeitung in Forschungsansätze, Dokumentation und Präsentation wissenschaftlicher Arbeiten und Ergebnisse).

Hauptstudium (77h + DA)5. Semester

<i>Softwarekonstruktion</i>	2V 2Ü
<i>Eingebettete Systeme</i>	4V 2Ü
<i>Darstellung, Verarbeitung und Erwerb von Wissen</i>	4V 2Ü
<i>BWL: Marketingforschung</i>	4V
	14V 6Ü 20h

Das 5. Semester als 1. Hauptstudiumssemester sollte der Orientierung im Hauptstudium dienen und sich auf Pflicht- und Wahlpflichtvorlesungen konzentrieren.

Zu den Informatik-Vorlesungen werden die Übungen besucht und die Übungsaufgaben werden zur Selbstkontrolle bearbeitet.

Weiterhin wird eine Veranstaltung des Nebenfachs absolviert mit dem Ziel, das Nebenfachstudium zügig abzuschließen.

Sollten sich im Grundstudium Verzögerungen eingestellt haben, können sie durch verstärkten Einsatz im 5. Semester ausgeglichen werden.

Ein Leistungsnachweis (Schein) wird erworben im Fach:

- *Softwarekonstruktion.*

Es werden folgende Prüfungen zu den Wahlpflichtvorlesungen und in der Nebenfachveranstaltung abgelegt:

- *Eingebettete Systeme,*
- *Darstellung, Verarbeitung und Erwerb von Wissen,*
- *BWL: Marketingforschung.*

Es wird empfohlen, nach Möglichkeit noch eine weitere Wahlpflichtvorlesung zur verbesserten Orientierung zu besuchen.

6. Semester

<i>Informatik und Gesellschaft</i>	1V 2Ü
<i>Übersetzerbau</i>	2V 2Ü
<i>Modellgestützte Analyse und Optimierung</i>	4V 2Ü
<i>Einführung ins funktionale Programmieren</i>	2V
<i>BWL: Beschaffungs- und Materialwirtschaft</i>	4V
	13V 6Ü 19h

Im 6. Semester sollen die noch fehlenden Pflicht- und Wahlpflichtvorlesungen absolviert werden. Zusätzlich sollte man auch schon eine Spezialvorlesung besuchen.

Zur Selbstkontrolle werden die Übungsaufgaben der Pflicht- und Wahlpflichtvorlesungen bearbeitet.

Leistungsnachweise (Scheine) werden in folgenden Fächern erworben:

- *Informatik und Gesellschaft,*
- *Übersetzerbau,*
- *Einführung ins funktionale Programmieren.*

Es werden folgende Prüfungen zur noch fehlenden dritten Wahlpflichtvorlesung und einer weiteren Nebenfachveranstaltung abgelegt:

- *Modellgestützte Analyse und Optimierung,*
- *BWL: Beschaffungs- und Materialwirtschaft.*

Mit Ende des 6. Semesters sollte die Entscheidung für das Schwerpunktgebiet gefallen sein. Unser beispielhaft Studierender wurde durch die Wahlpflichtvorlesung *Modellgestützte Analyse und Optimierung* ange-regt, das Schwerpunktgebiet *Rechnerarchitektur, Eingebettete Systeme und Simulation* zu wählen.

7. Semester

<i>PG</i>	8PG
<i>Kapazitätsplanung und Leistungsbewertung verteilter Systeme</i>	3V 1Ü
<i>Modellierung und Simulation diskreter und kontinuierlicher Systeme</i>	2V
<i>Seminar</i>	2S
<i>BWL: Integrierte betriebliche Informationssysteme</i>	4V
	9V 1Ü 8PG 2S 20h

Mit dem 7. Semester beginnt das Studium im gewählten Schwerpunktgebiet. Auch das Seminar wird aus dem Angebot des Schwerpunktgebiets gewählt, so dass sich ein Umfang von 8 SWS an Schwerpunktgebietsveran-staltungen ergibt.

Weiteren großen Raum nimmt die Projektgruppe ein. Unser Studierender wählt eine Gruppe, welche nicht unbedingt direkt im Schwerpunktgebiet liegt, welche ihn aber thematisch sehr interessiert, und in welcher die Gelegenheit besteht, sie gemeinsam mit befreundeten Studierenden zu besuchen.

Schließlich wird auch das Nebenfachstudium vorangetrieben.

Ein Leistungsnachweis (Schein) wird erworben zur Veranstaltung:

- *Seminar.*

Es werden folgende Prüfungen im Schwerpunktgebiet und der letzten Nebenfachveranstaltung abgelegt:

- *Kapazitätsplanung und Leistungsbewertung verteilter Systeme,*
- *Modellierung und Simulation diskreter und kontinuierlicher Systeme,*
- *BWL: Integrierte betriebliche Informationssysteme.*

Mit der bestandenen Nebenfachprüfung ist das Nebenfachstudium abgeschlossen. Der Studierende kann sich auf die Abrundung seines Informatikstudiums konzentrieren.

8. Semester

<i>PG (Fortsetzung)</i>	8PG
<i>Verteilte numerische Algorithmen</i>	2V 2Ü
<i>Ausgewählte Fragen der Sicherheit</i>	2V 2Ü
<i>Directed Model Checking</i>	2V
	6V 4Ü 8PG 18h

Das 8. Semester rundet das Informatikstudium durch den Abschluss der Projektgruppe, sowie noch fehlende Veranstaltungen des Schwerpunktgebiets und weitere Veranstaltungen des Wahlbereichs ab.

Leistungsnachweise (Scheine) werden in folgenden Veranstaltungen erworben:

- *PG,*
- *Ausgewählte Fragen der Sicherheit.*

Es werden folgende Prüfungen im Schwerpunktgebiet abgelegt:

- *Verteilte numerische Algorithmen,*
- *Directed Model Checking.*

Im Hinblick auf den „geringen“ Umfang von 18 SWS wird empfohlen, außer den hier aufgeführten durchaus auch noch die eine oder andere Veranstaltung mehr zu besuchen.

Weiterhin sollte man das 8. Semester nutzen, um die passende Diplomarbeit zu suchen, so dass man pünktlich im 9. Semester mit ihrer Bearbeitung beginnen kann.

9. Semester

<i>Diplomarbeit</i>	20DA
	20DA 20h

Mit dem Ende des 9. Semesters ist die Diplomarbeit als letzte Prüfungsleistung erbracht. Das Studium konnte mit einem Umfang von insgesamt 155 Wochenstunden (+Diplomarbeit) in der Regelstudienzeit von 9 Semestern absolviert werden.

Vorlesungszyklen

Die folgenden Tabellen enthalten den vorläufigen Stand der Planungen der Pflicht- und Wahlpflichtveranstaltungen der Diplom- und Bachelorstudiengänge für die kommenden Semester. Studentinnen und Studenten können diese Tabellen bei ihren Planungen berücksichtigen.

Änderungen vorbehalten!

Pflicht	SS 2008	WS 2008/2009
DAP I		Doberkat
DAP II	N. N.	
SWT		N. N.
RS		Marwedel/Jansen, Th.
BS + RvS I	Krumm	
BS + RvS II		Krumm
IS	Biskup	
GTI	Jansen, Th.	
TI f. AI	Bollig	
Logfi		Schwentick
I & G		

Wahlpflicht	SS 2008	WS 2008/2009
SWK	Doberkat	
ÜB		Padawitz
MMI		Müller
RSYS	Fink	
ES		Marwedel
MAO	Buchholz	
EA + KT	Wegener	Wegener
DVEW		Kerns-Isberner
FMSE		Steffen

Lehramt	SS 2008	WS 2008/2009
EDID		Vahrenhold
DID II	Vahrenhold	

Abkürzungsverzeichnis

DAP	Datenstrukturen, Algorithmen und Programmierung
SWT	Softwaretechnik
RS	Rechnerstrukturen
BS + RvS	Betriebssysteme, Rechnernetze und verteilte Systeme
IS	Informationssysteme
GTI	Grundbegriffe der Theoretischen Informatik
TI f. AI	Theoretische Informatik für Angewandte Informatik
I & G	Informatik und Gesellschaft
SWK	Software-Konstruktion
ÜB	Übersetzerbau
MMI	Mensch-Maschine Interaktion
RSYS	Rechensysteme
ES	Eingebettete Systeme
MAO	Modellgestützte Analyse und Optimierung
EA + KT	Effiziente Algorithmen und Komplexitätstheorie
DVEW	Darstellung, Verarbeitung, Erwerb von Wissen
FMSE	Formale Methoden des Systementwurfs
EDID	Einführung in die Didaktik der Informatik
DID	Didaktik der Informatik
Logfi	Logik f. Informatiker

Zuordnung Lehrveranstaltungen zu Schwerpunktgebieten

Prüfungsgebiete der Hochschullehrer des Fachbereichs Informatik (zzgl. eigene Vorlesungen)

Prüfungsgebiete	Prog. (KI + AI)	RS (KI)	DS (KI+AI)	GTI (KI)	DID	TheoProg (KI)	LogSys d. Informatik (KI)	Theorie d. Logikentwurfs (KI)	KT (KI)	EA (KI)	BS (KI+AI)	GS (KI+AI)	IS (KI+AI)	KI (KI+AI)	ProgSpr. u. ihre Übersetzer (KI+AI)	RvS (KI+AI)	Rechensysteme (KI+AI)	SWT (KI+AI)	Systemanalyse (AI)	Operations Research (KI+AI)	Simulation (KI+AI)	Rechnergest. Entwurf / Produktion	Prozessrechnertechnik (AI)	Wissensentdeckung in Datenbanken
	VD					HD																		
Biskup													*											
Bollig 1)			*				*																	
Buchholz																*				*				
Doberkat																	*							
Dittrich																								
Fink																	*							
Jansen 1)			*						*															
Kern-Isberner																								
Krumm 1)											*					* ₁₎								
Marwedel4)		*									* ₄₎						*				*	*		
Moraga																								
Morik 7) 8)	* ₇₎												* ₈₎											*
Müller 5)	*	*	*								* ₅₎	*												
Mutzel			*							*														
Padawitz	*		*			*	*								*									
Rahmann																								
Rehof	*		*			*											*							
Rudolph	*																		*					
Sauerhoff 1)			*				*	*	*															
Schwentick			*					*	*															
Sieling 1)			*				*	*	*															
Spinczyk																								
Steffen	*		*			*									*									
Vahrenhold 1)			* ₁₎		*				*															
Wegener			*	*			*	*	*											*				

Nach Absprache sind häufig weitere Prüfungsgebiete möglich: 1) Alle von mir gehaltenen Vorlesungen; weitere u. U. nach Absprache; 2) soweit SCHEINE als Grundlage; 3) Kombination mit anderen Stammvorlesungen nach Absprache; 4) nur in Kombination mit RA oder ES; 5) BS nur in Kombination mit GS; die Kombination von GS mit anderen Vorlesungen ist nicht ausgeschlossen; 6) Kombination mit anderen praktischen Stammvorlesungen; 7) nur für Java; 8) nur in Kombination mit KI

Sprechzeiten Professorinnen/Professoren, Habilitierte

Name	Gebäude, Raum	Telefon	Sprechstunde
Biskup, Joachim	GB V, R. 422	755-2569	Mi, 10.00 – 12.00 Uhr u. n. V.
Bollig, Beate	OH 14, 314	755-2598	bei Anwesenheit
Buchholz, Peter	GB V, R. 406 a	755-4746	Do, 10.00 - 11.30 Uhr u. n. V.
Dittrich, Gisbert	OH 16, R. 217	755-6444	Do, 10.30 - 11.30 Uhr
Doberkat, Ernst-Erich	GB IV, R. 314	755-2780	n. V.
Fink, Gernot	OH 16, R. E23	755-6151	Di, 11.00 - 12.00 Uhr
Jansen, Thomas	OH 14, R. 107	755-4702	Di, 12.30 - 13.30 Uhr u. n. V.
Kalkbrenner, Gerrit	OH 16, R. 105	755-6121	n. V.
Kern-Isberner, Gabriele	GB V, R. 416	755-2045	Do, 14.00 - 15.00 Uhr
Krumm, Heiko	GB V, R. 406b	755-4674	Do, 16.00 - 17.00 Uhr
Marwedel, Peter	OH 16, R. E21	755-6111	Mo, 13.00 – 14.00 Uhr
Morik, Katharina	GB IV, R. 115	755-5100	Mi, 10.00 - 12.00 Uhr
Müller, Heinrich	OH 16, R. 124	755-6324	Di, 10.30 - 11.30 Uhr
Mutzel, Petra	OH 14, R. 231	755-7700	Di, 14.15 - 15.15 Uhr
Padawitz, Peter	OH 16, 216	755-5108	Di, 16.00 - 17.00 Uhr
Rahmann, Sven	OH 14, R.	755-7701	
Rehof, Jakob	OH 14, R. 114	755-7751	Mo, 13.00 - 14.00 Uhr
Rudolph, Günter	OH 14, R. 232	755-7702	Di, 10.30 - 11.30 Uhr
Sauerhoff, Martin	OH 14, R. 332	755-5174	bei Anwesenheit
Schwentick, Thomas	OH 16, R. 214	755-6341	Di, 13.00 - 14.00 Uhr u. n. V.
Sieling, Detlef	OH 14, R. 340	755-2067	n. V.
Spinczyk, Olaf	OH 16, E01	755-6322	n. V.
Steffen, Bernhard	OH 14, R. 102	755-5800	n. V.
Vahrenhold, Jan	OH 14, R. 212	755-7711	Mo, 14.00 - 15.00 Uhr u. n. V.
Wegener, Ingo	OH 14, R. 302	755-2776	bei Anwesenheit

GB IV = Geschoßbau IV, Campus Süd (Baroper Str. 301)
 GB V = Geschoßbau V, Campus Süd (August-Schmidt-Str. 12)
 OH 14 = Otto-Hahn-Str. 14, Campus Nord (Technologiepark)
 OH 16 = Otto-Hahn-Str. 16, Campus Nord (Technologiepark)

Wichtige Sprechzeiten WS 2007/2008

		Gebäude, R.	Telefon	Email	Sprechstunde
Dekanat		OH 14, R. E12 OH 14, R. E06 OH 14, R. E12	755-2009 755-2121 755-2759	martina.gentzer@udo.edu jutta.kossmann@udo.edu simone.reichel@udo.edu	Mo – Fr, jeweils 8.30 – 15.00 Uhr
Skriptenverkaufsstelle		EF 50, R. 0.431	755-2062		Mo - Fr, 9.00 - 12.00 Uhr
Zentrum für Studienan- gelegenheiten (ZfS)		EF 66, R. EG 02	755-2138 755-3723	(foerster,hohmann) @verwaltung.uni.dortmund.de	Mo/Do, 9.00 - 12.00 Uhr Mi, 13.00 - 15.30 Uhr
Studiendekan	Prof. Dr. Schwentick	OH 16, 214	755-6341	schwentick@cs.uni-dortmund.de	Di, 13.00 - 14.00 Uhr
Studienberatung				studienberatung@cs.uni-dortmund.de	
	Esmeray Eskin	GB IV, R. 306	755-2613	eskin@studium.cs.uni-dortmund.de	nach Vereinbarung
	Jan-Hendrik Lochner	GB V, R. 409	755-5669	lochner@studium.cs.uni-dortmund.de	nach Vereinbarung
	Robin Nunkesser	OH 14, R. 333	755-5132	nunkesser@studium.cs.uni-dortmund.de	nach Vereinbarung
	Fachschaft	OH 14	755-2048	fachschaft@studium.cs.uni-dortmund.de	
Studienberatung LA	Prof. Dr. Vahrenhold	OH 14, 212	755-7711	vahrenhold@cs.uni-dortmund.de	Mo, 14.00 - 15.00 Uhr
Ausländerbeauftragte	Hubert Wagner	OH 16, 219	755-5107	auslaender@studium.cs.uni-dortmund.de	nach Vereinbarung
Auslandsstudium	Hans Decker	OH 14, R. E08	755-2208	hans.decker@udo.edu	bei Anwesenheit
Frauenbeauftragte	Doris Schmedding	GB IV, R. 307b	755-2436	doris.schmedding@udo.edu	Mo, 9.00 - 11.00 Uhr
Softwarepraktikum	Doris Schmedding	GB IV, R. 307b	755-2436	doris.schmedding@udo.edu	Mo, 9.00 - 11.00 Uhr
Hardwarepraktikum	Karl-Heinz Temme	OH 16, R. E24	755-6373	karl-heinz.temme@udo.edu	nach Vereinbarung
Software- Technologielabor	Stefan Dißmann	GB IV, R. 312	755-2482	stefan.dissmann@udo.edu	bei Anwesenheit
IRB	Eckard Schulte	GB V, R. 322	755-2429	eckard.schulte@cs.uni-dortmund.de	bei Anwesenheit
Studentenaccounts		GB V, R. 312	755-2789		siehe Aushang

Studienfachberatung des Fachbereichs Informatik