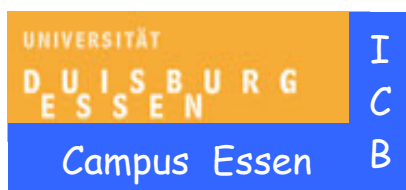


Modulhandbuch für das Lehramtsstudium Informatik

Module des Grundstudiums (FS 1 – FS 4) für das Lehramt „Unterrichtsfach Informatik“ mit dem Abschluss „Erste Staatsprüfung für das Lehramt an Gymnasien und Gesamtschulen“

Stand: 7.3.2005, überarbeitet am 10.5.2005



**Institute for Computer Science and
Business Information Systems (ICB)**
University of Duisburg-Essen
Schützenbahn 70, 45127 Essen
www.informatik.uni-essen.de

Inhalt

1. Einführung	3
1.1. Module	3
1.2. Leistungspunkte	3
1.3. Studienbegleitende Prüfungen	3
1.4. Prüfungsformen	3
1.5. Verwendung der Module in anderen Studiengängen	3
1.6. Redaktionelle Hinweise	4
2. Module des Grundstudiums	5
2.1. Mathematik 1 für Systems Engineering (9 CP)	5
06 xx xx xxx Vorlesung „Mathematik 1 (für Systems Engineering)“ (4 SWS, 6 CP).....	5
06 xx xx xxx Übung „Mathematik 1 (für Systems Engineering)“ (2 SWS, 3 CP).....	5
2.2. Mathematik 2 (für Systems Engineering) (9 CP)	6
06 xx xx xxx Vorlesung „Mathematik 2 (für Systems Engineering)“ (4 SWS, 6 CP).....	6
06 xx xx xxx Übung „Mathematik 2 (für Systems Engineering)“ (2 SWS, 3 CP).....	6
2.3. Modelle der Informatik 1 (9 CP)	7
05 31 xx xxx Vorlesung „Modelle der Informatik 1“ (4 SWS, 6 CP)	7
05 31 xx xxx Übung „Modelle der Informatik 1“ (2 SWS, 3 CP).....	7
2.4. Modelle der Informatik 2 (9 CP)	9
05 31 xx xxx Vorlesung „Modelle nebenläufiger Systeme“ (2 SWS, 3 CP, neu ab SS05)	9
05 31 xx xxx Vorlesung „Nebenläufiges Rechnen“ (2 SWS, 3 CP)	9
05 31 xx xxx Übung „Nebenläufiges Rechnen“ (2 SWS, 3 CP)	10
2.5. Anteil am fächerübergreifenden Didaktik-Modul im Grundstudium (9 CP)	9
05 31 xx xxx xxxxxxxx „Praxis der Fachdidaktik Informatik“ (2 SWS, 3 CP)	9
2.6. Programmierung (9 CP)	12
05 31 xx xxx Vorlesung „Programmierung“ (4 SWS, 6 CP)	12
05 31 xx xxx Übung „Programmierung“ (2 SWS, 3 CP unbenotet).....	12
2.7. Software Entwicklung & Programmierung (SEP) (3 CP)	13
05 31 xx xxx Praktikum „Software Entwicklung & Programmierung (SEP)“ (2 SWS, 3 CP).....	13

1. Einführung

1.1. Module

Unter Modularisierung versteht man die Zusammenfassung von Stoffgebieten zu thematisch und zeitlich abgerundeten, in sich geschlossenen und mit Leistungspunkten versehenen abprüfbaren Einheiten. Module können sich aus verschiedenen Lehr- und Lernformen zusammensetzen und Inhalte eines einzelnen Semesters oder eines Studienjahres umfassen, sich aber auch über mehrere Semester erstrecken. Wenn alle zu einem Modul gehörigen Prüfungsleistungen erbracht sind, werden dem Prüfungskonto Leistungspunkte gutgeschrieben und es wird die Note des Moduls berechnet.

1.2. Leistungspunkte

Das sog. Leistungspunktesystem (Credit Point System) dient der Erfassung der von den Studierenden erbrachten Leistungen sowie der Anrechnung von Prüfungsleistungen aus anderen Studiengängen. Auf der Grundlage von erworbenen Leistungspunkten (Credit Points) und der dabei erzielten Noten werden die gewichteten Durchschnittsnoten der Module und dadurch die Noten der Bachelor- bzw. Master-Prüfung insgesamt berechnet. Das Leistungspunktesystem folgt dem sog. ECTS-System. Beachten Sie, dass in diesem Handbuch statt Leistungspunkten auch der Begriff „Credit Points“ verwendet wird. In anderen Dokumenten werden auch die Begriffe „Anrechnungspunkte“, „Credits“, „ECTS-Credits“ und „ECTS-Credit Points“ verwendet.

Jede Lehrveranstaltung ist mit Leistungspunkten (Credit Points) versehen, die dem jeweils erforderlichen Studienaufwand (Workload) entsprechen. Ein Leistungspunkt entspricht dabei einem Studienaufwand von 30 Stunden effektiver Studienzeit; dies umfasst Präsenzzeiten, Vor- und Nachbereitung sowie Prüfungsvorbereitung. Der Umfang von Lehrveranstaltungen und die zugehörigen Leistungspunkte der einzelnen Lehrveranstaltungen sind in den Modulbeschreibungen festgelegt.

1.3. Studienbegleitende Prüfungen

Die Prüfungen sind studienbegleitend, d.h. ein Modul wird durch eine (oder mehrere) Prüfung(en) abgeschlossen, die spätestens im Anschluss an die letzte Veranstaltung des Moduls angeboten wird. Außerdem können - je nach Modul - für einzelne Lehrveranstaltungen innerhalb eines Moduls Leistungspunkte erworben werden, die in der Regel benotet sind. Ob ein Modul insgesamt oder durch Einzelprüfungen geprüft wird, wird durch den Dozenten bzw. die Dozentin festgelegt und ist in der Modulbeschreibung angegeben. Eine Gutschrift der Leistungspunkte für ein Modul erfolgt erst dann, wenn alle zu dem Modul gehörigen Leistungspunkte erworben wurden. Der Studien- und Prüfungsausschuss kann von diesem Schema abweichen und flexiblere Möglichkeiten für die Gutschrift von Leistungspunkten zulassen.

1.4. Prüfungsformen

Die Prüfungsform kann, abhängig von der Teilnehmerzahl, jeweils zu Semesterbeginn neu festgelegt werden. Bei kleinen Teilnehmerzahlen kann der Dozent statt schriftlicher Prüfungen (Klausuren) auch mündliche Prüfungen durchführen. Für die Prüfungen gibt es jeweils Haupt- und Nachtermin; der Haupttermin liegt in der Regel kurz nach Abschluss der zugehörigen Veranstaltung, der Nachtermin ca. 6-8 Wochen später, vor Beginn der Vorlesungen des nachfolgenden Semesters. Für Übungen und Praktika können auch unbenotete Leistungspunkte vergeben werden. Genauere Prüfungsmodalitäten sind in den Modulbeschreibungen und in der Prüfungsordnung festgelegt.

1.5. Verwendung der Module in anderen Studiengängen

Die Lehrveranstaltungen werden auch für andere Studiengänge verwendet. Leistungsnachweise und Prüfungsleistungen werden nach den entsprechenden Prüfungsordnungen vergeben. Bei der Vergabe von Leistungsnachweisen und Prüfungsleistungen wird hier je nach gültiger Prüfungsordnung ggf. in SWS oder in „alten“ Kreditpunkten gerechnet.

1.6. Redaktionelle Hinweise

Dieses Modulhandbuch wird im Jahresturnus aktualisiert und den Studierenden auf der ICB-Homepage <http://www.informatik.uni-essen.de/> als pdf-Dokument zur Verfügung gestellt.

Die Veranstaltungsnummern sind nicht stabil und daher nur ausnahmsweise aufgenommen. Durch die noch andauernde Neuordnung der Universität Duisburg-Essen ist auch eine Änderung der Ordnungssystematik zu erwarten.

Dieses Modulhandbuch dient sowohl als Informationsquelle für Studierende als auch als Dokument im Akkreditierungsprozess.

2. Module des Grundstudiums

2.1. Mathematik 1 für Systems Engineering (9 CP)

		Modulname: Mathematik 1 (für Systems Engineering)		
Studiengänge: Systems Engineering (BSc) Wirtschaftsinformatik (BSc) Lehramt Informatik		Veranstaltungstyp: Vorlesung 4 SWS Übung 2 SWS (mehrere Gruppen)		
Anzahl der CP: 9	Stud. Workload: 270 Stunden	Anzahl der SWS: 6 SWS	Modus: Pflichtmodul	Turnus: WS

06 xx xx xxx Vorlesung „Mathematik 1 (für Systems Engineering)“ (4 SWS, 6 CP)

Inhalt:

1. Grundlagen: Mengen, Relationen, Abbildungen, Beweisprinzipien, Zahlen, Zahlssysteme, Ungleichungen, Kombinatorik
2. Algebraische Strukturen: Gruppen, Ringe, Körper, Vektorräume
3. Lineare Algebra: Lineare Abbildungen, Matrizen, lineare Gleichungssysteme, Skalarprodukt, Orthogonalität, symmetrische Matrizen, Eigenwerte und Eigenvektoren
4. Einige geometrische Objekte: Geraden, Ebenen, Kreise, Ellipsen, Hyperbeln
5. Einige elementare Funktionen: Polynome, rationale Funktionen, Winkelfunktionen, Exponentialfunktionen und Logarithmen

Lernziele: Die Teilnehmer verstehen Konzepte, Notationen, Definitionen und Theoreme der genannten Gebiete.

06 xx xx xxx Übung „Mathematik 1 (für Systems Engineering)“ (2 SWS, 3 CP)

Inhalt: Die Übungen finden in Kleingruppen statt. Der Stoff der Vorlesung wird durch aktive Beschäftigung mit wöchentlich gestellten Übungsaufgaben vertieft.

Lernziele: Die Teilnehmer lernen Konzepte, Notationen, Definitionen und Theoreme der genannten Gebiete aktiv anzuwenden.

Literatur:

- G. Fischer: Lineare Algebra; Vieweg Verlag
- K. Endl: Analytische Geometrie und lineare Algebra; VDI Verlag
- L. Papula: Mathematik für Ingenieure; Band 1 und 2, Vieweg Verlag (dies ist die primäre Referenz)
- P. Furlan: Das gelbe Rechenbuch; Verlag Marita Furlan
- Aus dem Verlag Harri Deutsch: Mathematik, Bände 1, 2, 4, 5 (und 17)

Erforderlich: Mathematische Grundausbildung auf Schulniveau (Gymnasiale Oberstufe).

Nützlich: Es wird dringend die Teilnahme an einem „Mathematischen Vorkurs“ empfohlen. In der Regel dauert ein mathematischer Vorkurs vier Wochen und hat einen Gesamtumfang von 80 Stunden, je zur Hälfte als Vorlesung und als Übung. Beginn ist jeweils Anfang September, nähere Informationen finden sich auf der Homepage der Mathematik <http://www.uni-essen.de/fb6/>

Prüfungsmodalitäten: Schriftliche Prüfung (Klausur) über 9 CP. Es werden jeweils Haupt- und Nachtermin angeboten.

Autor/Dozent: im WS 04/05: Prof. Dr. Dieter Lutz; die Veranstaltung wird im Jahresrhythmus von verschiedenen Dozenten der Mathematik gehalten.

Homepage der Mathematik: <http://www.uni-essen.de/fb6/>

Letzte Änderungen: 18.2.2004, 30.8.2004, 7.3.2005

[Zurück zum Inhaltsverzeichnis](#)

2.2. Mathematik 2 (für Systems Engineering) (9 CP)

		Modulname: Mathematik 2 für Systems Engineering		
Studiengänge: BSc Systems Engineering BSc Wirtschaftsinformatik Lehramt Informatik		Veranstaltungstyp: Vorlesung 4 SWS Übung 2 SWS (mehrere Gruppen)		
Anzahl der CP: 9	Stud. Workload: 270 Stunden	Anzahl der SWS: 6 SWS	Modus: Pflichtmodul	Turnus: SS

06 xx xx xxx Vorlesung „Mathematik 2 (für Systems Engineering)“ (4 SWS, 6 CP)

Inhalt:

1. Grenzwerte: Folgen, Reihen, Grenzwerte, stetige Funktionen
2. Differentialrechnung: Ableitungsregeln, Extremwerte, Potenzreihenentwicklung, Nullstellenbestimmung
3. Integralrechnung: Stammfunktion und Flächenintegral, Integrationsregeln, uneigentliche Integrale
4. Mehrdimensionale Analysis: Partielle und totale Ableitung, Extremwerte, Kurven, Mehrfachintegrale

Lernziele: Die Teilnehmer verstehen Konzepte, Notationen, Definitionen und Theoreme der genannten Gebiete.

06 xx xx xxx Übung „Mathematik 2 (für Systems Engineering)“ (2 SWS, 3 CP)

Inhalt: Die Übungen finden in Kleingruppen statt. Der Stoff der Vorlesung wird durch aktive Beschäftigung mit wöchentlich gestellten Übungsaufgaben vertieft.

Lernziele: Die Teilnehmer lernen Konzepte, Notationen, Definitionen und Theoreme der genannten Gebiete aktiv anzuwenden.

Literatur:

- G. Fischer: Lineare Algebra; Vieweg Verlag
- K. Endl: Analytische Geometrie und lineare Algebra; VDI Verlag
- L. Papula: Mathematik für Ingenieure; Band 1 und 2, Vieweg Verlag
- P. Furlan: Das gelbe Rechenbuch; Verlag Marita Furlan
- Aus dem Verlag Harri Deutsch: Mathematik, Bände 1, 2, 4, 5 (und 17)

Erforderliche Module: Mathematik 1 für Systems Engineering

Prüfungsmodalitäten: Schriftliche Prüfung (Klausur) über 9 CP. Es werden jeweils Haupt- und Nachtermin angeboten.

Autor/Dozent: im SS 05: Prof. Dr. Dieter Lutz; die Veranstaltung wird im Jahresrhythmus von verschiedenen Dozenten der Mathematik gehalten.

Homepage der Mathematik: <http://www.uni-essen.de/fb6/>

Letzte Änderungen: 18.2.2004, 12.7.2004, 30.8.2004, 7.3.2005

[Zurück zum Inhaltsverzeichnis](#)

2.3. Modelle der Informatik 1 (9 CP)

					Modulname: Modelle der Informatik 1				
Studiengänge: Systems Engineering (BSc), Wirtschaftsinformatik (BSc) Lehramt Informatik					Veranstaltungstyp: Vorlesung 4 SWS Übung 2 SWS (mehrere Gruppen)				
Anzahl der CP: 9		Stud. Workload: 270 Stunden		Anzahl der SWS: 6 SWS		Modus: Pflichtmodul		Turnus: WS	

05 31 xx xxx Vorlesung „Modelle der Informatik 1“ (4 SWS, 6 CP)

Inhalt:

1. Formale Sprachen: Buchstaben, Wörter, Sprachen, Klassen von unendlichen Sprachen, Grammatiken: Definitionen, Chomsky-Hierarchie, BNF, EBNF, Endliche Automaten und reguläre Sprachen: Moore- und Mealy-Automaten, Deterministische und Nichtdeterministische Automaten, reguläre Sprachen; Kontextfreie Sprachen, Ableitungsbäume, Scanner und Parser; Beispiele: HTML, XML.
2. Logik: Aussagenlogik, logische Ausdrücke und Wahrheitstabellen, Tautologien, de Morgansche Regeln; Beweismethoden, aussagenlogische Resolution, Normalformen, Resolvierung von Begründungen; Grundzüge der Prädikatenlogik, Einführung in die Temporale Logik.
3. Bäume, Graphen und Netzwerke: Definitionen von Bäumen, binäre Suchbäume, Baumdurchlauf, ausgeglichene Bäume, Mehrwegbäume, Exkurs über Hashverfahren; Definitionen von Graphen, Euler- und Hamilton-Graphen; Knotenfärbung; Schwacher und starker Zusammenhang, Tiefen- und Breitendurchlauf, Spannbäume, Minimale Spannbäume, kürzeste Wege (Dijkstra-Algorithmus); Anwendungen, z.B. Routing in Rechnernetzen; Netzwerke und Flüsse.
4. Petri-Netze: Definition von Petri-Netzen, Stellen/Transitionsnetze, Lebendigkeit, Beschränktheit, S- und T-Invarianten, Erreichbarkeit; Modelle für wechselseitigen Ausschluss, Produzent/Konsument-Problem und Leser/Schreiber-Problem; Bedingungs/Ereignisnetze, Farbige Petri-Netze, Petri-Netze mit Verbotskanten; Vergrößerung/Verfeinerung und Faltung/Entfaltung von Petri-Netzen; Ausblick auf stochastische Petri-Netze.
5. Stochastische Modelle: Überblick über Stochastische Petri-Netze, Zeitdiskrete Markovketten, Pseudo-Zufallszahlen und Monte-Carlo-Simulation.
6. Ausblick auf weitere Aspekte der theoretischen Informatik, z.B. Turingmaschinen, Berechenbarkeit, Komplexität und Effizienz von Algorithmen, Konzept der NP-Vollständigkeit.

Lernziele: Kennen lernen von Modellierungsparadigmen und Formalismen, die sich in der praktischen Anwendung als erfolgreich erwiesen haben; Erlernen der grundlegenden Konzepte, der zugehörigen Formalismen und Notationen, der Anwendungsbereiche und der wichtigsten Algorithmen.

05 31 xx xxx Übung „Modelle der Informatik 1“ (2 SWS, 3 CP)

Inhalt: Aufgaben und Beispiele zum Stoff der Vorlesung.

Lernziele: Weiterentwicklung von Problemlösungsfähigkeiten mit Hilfe von Modellen; Fähigkeit zur praktischen Anwendung von Methoden und Techniken der Modellierung und Analyse. Die Übungen finden in Gruppen statt. Es werden mehrere Termine angeboten.

Literatur und weiteres Material:

- B. Müller-Clostermann: Skriptum „Modelle der Informatik“ (siehe Homepage)
- U. Hedstück: Einführung in die Theoretische Informatik – Formale Sprachen und Automatentheorie, Oldenbourg, 2002 (176 Seiten), in ca. 50 Exemplaren in der Lehrbuchsammlung (am Campus Essen)
- U. Schöning: Theoretische Informatik – kurzgefasst, 2001 (4. Auflage, 198 Seiten)
- J. Kelley: Logik im Klartext, Pearson Studium, 2003, in ca. 50 Exemplaren in der Lehrbuchsammlung am Campus Essen
- B. Baumgarten: Petri-Netze: Grundlagen und Anwendungen; Spektrum-Akademischer Verlag, 1997

Erforderlich: Keine formalen Voraussetzungen; möglichst im FS 1 zu absolvieren.

Nützlich: Elementare Mathematikkenntnisse (Mengen, Abbildungen, Funktionen, Relationen; Matrizen und Vektoren, Skalarprodukt, Lineare Gleichungssysteme; Zufallsvariable, Verteilung)

Prüfungsmodalitäten: Klausur und Nachklausur über 9 CP (Februar und April)

Beiträge zu den Modulen: Grundlegend für die Module in den nachfolgenden Fachsemestern

Autor/Dozent:

Prof. Dr. Bruno Müller-Clostermann, FB5/ICB, bmc@informatik.uni-essen.de,

Homepage: <http://www.informatik.uni-essen.de/SysMod/lehre> (Skript und weiteres Begleitmaterial)

Weitere Dozenten:

Prof. Dr. Michael Goedicke, FB5/ICB, goedicke@informatik.uni-essen.de

Prof. Dr. Klaus Echtele, FB5/ICB, echtle@informatik.uni-essen.de

Letzte Änderungen: 17.2.2004, 3.7.2004, 7.3.2005

[Zurück zum Inhaltsverzeichnis](#)

2.4. Modelle der Informatik 2 (9 CP)

		Modulname: Modelle der Informatik 2		
Studiengänge: Systems Engineering (BSc) Lehramt Informatik		Veranstaltungstyp: 1 Vorlesung mit 2 SWS 1 Vorlesung mit 2 SWS und 1 Übung mit 2 SWS. Es werden mehrere Übungsgruppen angeboten.		
Anzahl der CP: 9	Stud. Workload: 270 Stunden	Anzahl der SWS: 6 SWS	Modus: Pflichtmodul	Turnus: SS

05 31 xx xxx Vorlesung „Modelle nebenläufiger Systeme“ (2 SWS, 3 CP, neu ab SS05)

Inhalt: In der Vorlesung "Modelle nebenläufiger Systeme" werden vor allem automatenbasierte Ansätze zur Spezifikation und Analyse nebenläufiger Systeme vorgestellt. Die Themen umfassen folgende Punkte:

1. Kommunizierende Automaten: synchrone und asynchrone Automatenkommunikation, Transitionssysteme, Verhaltensspezifikation mit Automaten, Weg/Zeit-Diagramme, Prinzipie der Protokollspezifikation und der automatenbasierten Erreichbarkeitsanalyse, Algorithmen zur Erreichbarkeitsanalyse, erschöpfende und partielle Exploration; probabilistische Exploration.
2. Übersicht über die Spezifikations- und Beschreibungssprache SDL
3. Einführung in Timed Automata (synchrone Automaten mit Zeit), Einführung in das Werkzeug UPPAAL, Erreichbarkeitsanalyse mit UPPAAL, Model Checking, Deadlock, Safety and Liveness Properties
4. Einführung in Prozessalgebra, Labeled Transition Systems, Einführung in das Werkzeug LTSA.

Die Veranstaltung "Modelle nebenläufiger Systeme" dient dazu, das Bachelor-Studium Systems Engineering mit weiteren formalen Grundlagen auszustatten.

Literatur

- B. Müller-Clostermann: Übersichtsfolien zu "Modelle nebenläufiger Systeme"; (ab SS 05 als pdf-Dokument erhältlich)
- UPPAAL-Tool und Tutorien [UPPAAL-Homepage: http://www.uppaal.com](http://www.uppaal.com)
- Tutorial on Uppaal. (Version 2004, vgl UPPAAL-Homepage)
- Uppaal2k: Small Tutorial. A short description of the tool as well as some examples. (vgl. UPPAAL-Homepage)
- J. Magee, J. Kramer: Concurrency - State models and Java programs; Wiley, 1999
- Das LTSA-Tool und englischsprachige Folien zu og. Lehrbuch gibt es unter <http://www-dse.doc.ic.ac.uk/concurrency>

Lernziele: Die Studierenden sollen ein tieferes Verständnis für die Modellierung nebenläufiger Systeme erhalten, im Vordergrund stehen insbesondere Spezifikations- und Analysetechniken für zustandsbasierte Systeme.

- **Hinweis:** Diese Veranstaltung wird erstmals ab dem SS 2005 angeboten. Sie dient dazu, das Bachelor-Studium Systems Engineering und das Studium des Unterrichtsfaches Informatik mit weiteren formalen Grundlagen auszustatten. Wesentliche Grundlage ist das Lehrbuch von J. Magee, J. Kramer: „Concurrency – State models and Java programs“, Wiley 1999 und die zugehörigen, frei verfügbaren Foliensätze.

Literatur:

- B. Müller-Clostermann: Übersichtsfolien zu „Modelle nebenläufiger Systeme“; siehe Homepage (ab SS 05 als pdf-Dokument erhältlich)
- J. Magee, J. Kramer: Concurrency – State models and Java programs; Wiley, 1999
- Die englischsprachigen Folien stehen zur Verfügung unter <http://www-des.doc.ic.ac.uk/concurrency>

05 31 xx xxx Vorlesung “Nebenläufiges Rechnen“ (2 SWS, 3 CP)

Inhalt: Die Vorlesung vermittelt die grundlegenden Konzepte der Nebenläufigkeit, die in der Hardware, im Betriebssystem, in Datenbanksystemen, in Programmen, in verteilten Systemen und in Rechnernetzen eine wichtige Rolle spielen, in einheitlicher Weise. Themen der Vorlesung sind:

1. Programmmodell, Ausführungsmodell, Ablaufmodell, Schreibkonflikt, Datenabhängigkeit, Verklemmung
2. Parallelisierung in der Hardware, Parallelisierung im Grundsystem, Parallelisierung auf Programmebene, Operatoren zur Synchronisation und Kommunikation
3. Mehrrechnersysteme, Mehrprozessbetrieb, Prozessplanung, Prozessumschaltung
4. Threads in Java, Interaktion zwischen Threads, Monitore, Wartebedingungen in Programmen, Vermeidung von Verklemmung, Grundmuster des nebenläufigen Rechnens für bestimmte Anwendungen, Echtzeitprozesse, Programmierstile für Echtzeitsysteme (zeitbasierte und ereignisorientierte Ansätze)

Lernziele: Nebenläufigkeit als ein allgemeines Prinzip begreifen, das viele Gebiete der Informatik durchzieht. Konstruktion und Analyse von nebenläufigen Systemen.

05 31 xx xxx Übung „Nebenläufiges Rechnen“ (2 SWS, 3 CP)

Inhalt: Erweiterung der Programmierfähigkeit von sequentiellen auf nebenläufige Programme.

Lernziele: Modellierung von nebenläufigen Systemen, Programmierung von nebenläufigen Threads.

Literatur:

- Begleitmaterial und Skripte, siehe Homepage der Autoren/Dozenten
- A. Tanenbaum: Moderne Betriebssysteme; Hanser-Verlag
- R. G. Hertzich, G. Hommel: Nebenläufige Programme; Springer-Verlag

Erforderliche Module: Programmierung, Modelle der Informatik 1

Prüfungsmodalitäten: Es wird eine Klausur über das gesamte Modul „Modelle der Informatik 2“ mit insgesamt 9 CP angeboten. Es wird einen Haupt- und einen Nachtermin geben. Für andere Studiengänge können ggf. andere Prüfungsmodalitäten gelten.

Autoren/Dozenten:

Modelle nebenläufiger Systeme: Prof. Dr. Bruno Müller-Clostermann, FB5/ICB, bmc@informatik.uni-essen.de

Nebenläufiges Rechnen: Prof. Dr. Klaus Echtele, FB5/ICB, echtle@informatik.uni-essen.de

zusammen mit Dr. Irene Eusgeld, FB5/ICB, eusgeld@informatik.uni-essen.de

Homepages: <http://www.informatik.uni-essen.de/SysMod/lehre> (BMC)
<http://dc.informatik.uni-essen.de> (Echtele)

Letzte Änderungen: 17.2.2004, 12.7.2004, 7.3.2005, 10.5.2005

[Zurück zum Inhaltsverzeichnis](#)

2.5. Anteil am fächerübergreifenden Didaktik-Modul im Grundstudium (9 CP)

		Modulname: Anteil am fächerübergreifenden Didaktik-Modul im Grundstudium		
Studiengänge: Lehramt Informatik		Veranstaltungstyp: ###		
Anzahl der CP: 3	Stud. Workload: 90 Stunden	Anzahl der SWS: 2 SWS	Modus: Pflichtmodul	Turnus: WS/SS (3./4. FS)

05 31 xx xxx ### „Praxis der Fachdidaktik Informatik“ (2 SWS, 3 CP)

Inhalt: noch unbekannt

Lernziele: noch unbekannt

Literatur:

- Noch unbekannt

Erforderliche Module: Keine

Prüfungsmodalitäten: nach Vereinbarung

Autoren/Dozenten:

Prof. Dr. U. Hoppe

Homepages: noch unbekannt

Letzte Änderungen: 14.3.2005

[Zurück zum Inhaltsverzeichnis](#)

2.6. Programmierung (9 CP)

					Modulname: Programmierung				
Studiengänge: Systems Engineering (BSc), Wirtschaftsinformatik (BSc) Lehramt Informatik					Veranstaltungstyp: Vorlesung 4 SWS Übung 2 SWS				
Anzahl der CP: 9		Stud. Workload: 270 Stunden		Anzahl der SWS: 6 SWS		Modus: Pflichtmodul		Turnus: WS	

05 31 xx xxx Vorlesung „Programmierung“ (4 SWS, 6 CP)

Inhalt: Es wird das strukturierte objektorientierte Programmieren mit der Programmiersprache Java vermittelt. Außerdem werden ausgewählte Algorithmen sowie Strategien zu deren Entwurf behandelt. Die Themen folgen den Kapiteln des vorgeschlagenen Lehrbuchs „Lehrbuch der Programmierung mit Java“:

1. Grundbegriffe der Informatik; Problemlösen durch Methoden und Maschinen der Informatik; Algorithmusbegriff, Bezüge zu Formalen Sprachen und Grammatiken.
2. Grundelemente der Programmierung; Primitive Typen, Anweisungen, Arrays.
3. Objekte und Klassen; Grundzüge der Objektorientierung, Verweisvariablen und Zugriffe auf Objekte, Methoden und ihre Parameter, Konstruktoren, Gültigkeitsbereich von Bezeichnern.
4. Rekursion; Beschreibung mit Selbstbezug, Rekursive Algorithmen, Rekursive Datenstrukturen, Arten rekursiver Beschreibungen.
5. Datenstrukturen; Zeichenkette, Puffer und Stapel, Suchbaum, Hashtabelle, Gerichteter Graph.
6. Erweiterung von Klassen; Erweiterung einer Klassenimplementierung und Erzeugung von Objekten, Verdecken von Variablen und Überschreibung von Methoden, Vererbungshierarchien, Anonyme Erweiterung von Klassen, Beziehungen zwischen Klassen.
7. Flexible Softwarekomponenten: Generische Objektstrukturen; Verwendung von Programmteilen, Abstrakte Klassen, Definition von Schnittstellen, Verwendung von Schnittstellen.
8. Spezielle Konzepte der Programmierung; Pakete, Ausnahmen, Threads.

Lernziele: Die Studierenden lernen die Grundelemente einer Programmiersprache sowie die wesentlichen Datenstrukturen und zugehörige Algorithmen kennen.

05 31 xx xxx Übung „Programmierung“ (2 SWS, 3 CP unbenotet)

Inhalt: Vertiefende Aufgaben und Beispiele zum Stoff der Vorlesung sowie praktische Übungen, wobei das aktive Programmieren im Vordergrund steht.

Lernziele: Fähigkeit zur Anwendung der in der Vorlesung vorgestellten Konzepte der verwendeten Programmiersprache; Vertiefung des Verständnisses der Konzepte durch ihre Anwendung.

Literatur:

- K. Echte, M. Goedicke: Lehrbuch der Programmierung mit Java; d-Punkt-Verlag
- K. Arnold, J. Gosling: The Java Programming Language; Addison-Wesley

Prüfungsmodalitäten: Für die Vorlesung: Schriftliche Prüfung über die Vorlesung im Umfang von 6 CP. Es werden zwei Prüfungstermine (Haupt- und Nachtermin) angeboten.

Für die Übungen: Prüfungen sind begleitend zur Übung; es werden jeweils 3 CP unbenotet vergeben.

Autoren/Dozenten:

Prof. Dr. Michael Goedicke, FB5/ICB, goedicke@cs.uni-essen.de

Prof. Dr. Klaus Echte, FB5/ICB, echtle@cs.uni-essen.de

(und weitere Professoren des ICB)

Homepage: <http://s3.cs.uni-essen.de/> (aktuelle Informationen, sowie Skripte im Semester)

Hinweis: Die Vorlesung Programmierung inkl. Übungen wird auch für andere Studiengänge angeboten. (Je nach PO-Version werden die Namen „Grundzüge der Informatik 1“ oder „Algorithmen und Datenstrukturen“ verwendet werden.)

Letzte Änderungen: 18.2.2004, 25.8.2004, 7.3.2005

[Zurück zum Inhaltsverzeichnis](#)

2.7. Software Entwicklung & Programmierung (SEP) (3 CP)

		Modulname: Software Entwicklung & Programmierung (SEP)		
Studiengänge: Systems Engineering (BSc), Wirtschaftsinformatik (BSc), Lehramt Informatik		Veranstaltungstyp: Übung 2 SWS		
Anzahl der CP: 3	Stud. Workload: 90 Stunden	Anzahl der SWS: 2 SWS	Modus: Pflichtmodul	Turnus: SS und WS

05 31 xx xxx Praktikum "Software Entwicklung & Programmierung (SEP)" (2 SWS, 3 CP)

Inhalt: Die Anwendung der in der Vorlesung Programmierung und Übung erworbenen Kenntnisse in kleinen bis mittelgroßen Projekten, die auch zum Teil in Gruppen von ca. 6 – 8 Teilnehmern durchgeführt werden sollen. Der Softwarelebenszyklus soll in wichtigen Stadien durchlaufen werden und die entsprechenden Dokumente (Anforderungsbeschreibung, Design und Implementierung) sollen erstellt und reviewed werden.

Lernziele: Programmieren im Team, Strukturierung von komplexen Softwaresystemen mit den Mitteln einer objektorientierten Programmiersprache. Verwendung einfacher Entwicklungsumgebungen.

Literatur und Materialien: Siehe SEP-Homepage www.icb.uni-essen.de/sep/

Prüfungsmodalitäten: Prüfungen sind begleitend zur Übung; es werden jeweils 3 CP unbenotet vergeben.

Autoren/Dozenten:

Prof. Dr. Klaus Pohl, FB5/ICB, pohl@sse.uni-essen.de

Prof. Dr. Michael Goedicke, FB5/ICB, goedicke@cs.uni-essen.de

Prof. Dr. Klaus Echte, FB5/ICB, echtle@cs.uni-essen.de

(und weitere Professoren des ICB)

Homepage: <http://s3.cs.uni-essen.de/> (aktuelle Informationen, sowie Skripte im Semester)

Letzte Änderungen: 18.2.2004, 25.8.2004, 7.3.2005

[Zurück zum Inhaltsverzeichnis](#)