

## Lehrinhalte des Bachelorstudiums Informatik (521)

an der Technisch-Naturwissenschaftlichen Fakultät  
der Johannes Kepler Universität Linz

gültig ab WS 2010/11

**Studiendauer:** 6 Semester  
**Abschluss:** Bachelor of Science (BSc)  
**Profil:** breit einsetzbare IT-Fachleute

### Fächer

	ECTS	Wochen- stunden
<i>Pflichtfächer</i>		
Propädeutikum	1,5	1
Theorie	36,0	24
Hardware	22,5	15
Software	31,5	21
Systeme	24,0	16
Anwendungen	22,5	15
Begleitende Inhalte	15,0	10
<i>Wahlfächer</i>		
Informatik-Wahlfach	7,5	5
Freie Lehrveranstaltungen	9,0	6
<i>Bachelorarbeiten</i>		
Projektpraktikum	7,5	5
Seminar	3,0	2
<b>Gesamt</b>	<b>180,0</b>	<b>120</b>

# Semesteraufteilung

## 1. Semester

Propädeutikum	1KV
Mathematische Grundlagen	2VO+2UE
Diskrete Strukturen	1VO
Digitale Schaltungen	2VO+1UE
Softwareentwicklung 1	2VO+2UE
Informationssysteme 1	2VO+2UE
Präsentations- u. Arbeitstechnik	2KV
Freie Lehrveranstaltungen	2

## 2. Semester

Algebra	3VO+2UE
Elektronik	2VO+1UE
Softwareentwicklung 2	2VO+2UE
Algorithmen u. Datenstrukturen 1	2VO+1UE
Betriebssysteme	2VO+1PR
Wirtschaftsgrundlagen für Informatiker	2VO

## 3. Semester

Analysis	2VO+2UE
Berechenbarkeit u. Komplexität	2VO+1UE
Computer Architecture 1	3VO+1UE
Algorithmen u. Datenstrukturen 2	2VO+1UE
Systemnahe Programmierung	2PR
Netzwerke und verteilte Systeme	2VO+1UE
Rechtsgrundlagen für Informatiker	2VO

## 4. Semester

Formal Models	2VO+1UE
Statistik	2VO+2UE
Computer Architecture 2	2VO+1UE
PR Digitale Schaltungstechnik	2PR
Praktikum Softwareentwicklung 2	2PR
Multimediasysteme	2VO+1UE
Ethik u. Gender Studies	2KV
Freie Lehrveranstaltungen	2

## 5. Semester

Software Engineering	2VO+1UE
Übersetzerbau	2VO+2UE
Informationssysteme 2	2VO+1UE
Artificial Intelligence	2VO+1UE
Bioinformatics	2VO
Projektorganisation	2KV
Bachelor-Seminar	2SE
Freie Lehrveranstaltungen	2

## 6. Semester

Embedded & Pervasive Systems	2VO+1UE
Computer Graphics	2VO+1UE
Informatik-Wahlfächer	5
Projektpraktikum	5PR

### Lehrveranstaltungsarten

VO	Vorlesung (z.B. 2VO bedeutet 2-stündige Vorlesung)
UE	Übung
KV	Kombinierte Veranstaltung (d.h. Vorlesung + Übung)
PR	Praktikum
SE	Seminar

### Institute

ALG	Institut für Algebra	FMV	Formale Modelle und Verifikation
BIO	Bioinformatik	PC	Pervasive Computing
CA	Computer Architecture	RIIC	Integrierte Schaltungen
CG	Computer Graphics	RISC	Symbolisches Rechnen
CP	Computational Perception	SEA	Systems Engineering und Automation
FAW	Anwendungsorientierte Wissensverarbeitung	SSW	Systemsoftware
FIM	Informationsverb. und Mikroprozessortechnik	TK	Telekooperation
FLL	Wissensbasierte Mathematische Systeme		

# 1. Semester

## **Propädeutikum** (1KV; SEA, Egyed)

*Ziele und Inhalt:* Die KV vermittelt einen Überblick über das Studium der Informatik und präsentiert grundlegende Konzepte der Informatik und ihrer Methoden (z.B. Algorithmenbegriff, Information, Softwareentwicklung, Parallelität, etc.).

## **Mathematische Grundlagen** (2VO + 2UE; FLLL, Klement)

*Ziele:* Erlernen grundlegender mathematischer Strukturen und formaler Systeme anhand verschiedener mathematischer Bereiche.

*Inhalt:* Schaltalgebra, Aussagenlogik, Mengenlehre, Prädikatenlogik, Datenstrukturen, allgemeine Relationen sowie Äquivalenz- und Ordnungsrelationen, Funktionen und im Speziellen reelle Funktionen.

## **Diskrete Strukturen** (1VO; FMV, FIM, Biere - Mühlbacher)

*Ziele:* Grundlagen der Zahlentheorie und Graphentheorie insbesondere für Algorithmen, die in den nachfolgenden Vorlesungen über Programmieren und Algorithmen und Datenstrukturen benötigt werden.

*Inhalt:* Zahlentheorie: Natürliche, rationale und reelle Zahlen; größter gemeinsamer Teiler, kleinstes gemeinsames Vielfaches; Primzahlen, Primfaktoren; lineare diophantische Gleichungen; Kongruenzen und Restklassensysteme. Graphentheorie: Gerichtete und ungerichtete Graphen, Hyper-Graphen, Wege, Kreise, Brücken, Zusammenhang und Komponenten, kürzeste Wege, Gerüste, Matching auf Paaren Graphen, Artikulationspunkte; Bäume, wichtige Baumklassen wie Fibonacci -Bäume und balancierte Bäume. Motivierende Anwendungsbeispiele.

## **Digitale Schaltungen** (2VO + 1UE; CA, Schneider)

*Ziele:* Einführung in die Grundlagen der Digitaltechnik. Schrittweise Erarbeitung des Entwurfs logischer Schaltungen.

*Inhalt:* Darstellung von Information (ganze Zahlen, Gleitkommazahlen, Texte), Schaltungen auf Gatter-Ebene (Boolesche Algebra, Schaltfunktionen, Schaltnetze, Minimierung von Schaltnetzen, Schaltwerke), Schaltungen auf Register-Transfer-Ebene (Komponenten, Design-Methoden).

## **Softwareentwicklung 1** (2VO + 2UE; PC, Ferscha)

*Ziele:* Einführung in moderne Methoden der Programmierung. Es wird Java verwendet, aber die Betonung liegt auf allgemein einsetzbaren Konzepten und Methoden der Softwareentwicklung.

*Inhalt:* Problemlösungsmethoden, Grundbegriffe der Programmierung, Variablen, Datentypen, Ausdrücke, Anweisungen, Ein-/Ausgabe, Objekte, Felder und Methoden, Vererbung, Interfaces, Dynamische Datenstrukturen, Rekursion, Ausnahmen, Threads, Pakete, Dokumentation und Programmierstil.

## **Informationssysteme 1** (2VO + 2UE; FAW, Wagner)

*Ziele:* Einführung in die fundamentalen Konzepte, Methoden und Sprachen des Datenbankentwurfs. Der Schwerpunkt liegt im konzeptuellen und logischen Bereich.

*Inhalt:* Anforderungserhebung und -analyse, konzeptueller Datenbankentwurf, logischer Datenbankentwurf, relationale Datenbanken, SQL, Transaktionen und Synchronisationsverfahren.

## **Präsentations- und Arbeitstechnik** (2KV; SEA, Grünbacher)

*Ziele:* Einführung in Methoden des wissenschaftlichen Arbeitens sowie Übung und Feedback beim Präsentieren eigener Arbeiten.

*Inhalt:* Literatursuche, selbständige Erarbeitung wissenschaftlicher Literatur, Abfassen eines schriftlichen "Papers" über das erarbeitete Thema, Vortrag über das Thema, Diskussion.

## 2. Semester

### **Algebra** (3VO + 2UE; ALG, Pilz)

*Ziele:* Einführung in die lineare Algebra.

*Inhalt:* Vektoren, Skalar- und Vektorprodukt, Gleichungssysteme, Vektorräume, Lineare Abhängigkeit, Basen, Algebraische Strukturen, Matrizen, Lineare Abbildungen, Basistransformationen, Affine Abbildungen, Determinanten, Eigenwerte.

### **Elektronik** (2VO + 1UE; RIIC, Hagelauer)

*Ziele:* Einführung in die Grundlagen der Elektronik und Analogtechnik für Informatiker.

*Inhalt:* Elektrischer Widerstand, Ohmsches Gesetz, Kirchhoffsche Gesetze, Berechnung von Widerstandsnetzwerken mit unabhängigen Quellen, Ladevorgänge an Kondensatoren und Spulen, Dioden, Transistoren, Beispiele mit Dioden und Transistoren.

### **Softwareentwicklung 2** (2VO + 2UE; SSW, Mössenböck)

*Ziele:* Festigen der Programmier-Kompetenz durch fortgeschrittene Methoden der Softwareentwicklung anhand objektorientierter Architekturen und der Java-Klassenbibliothek.

*Inhalt:* Grundbegriffe objektorientierten Denkens, Klassen, Vererbung, dynamische Bindung, Interfaces, Kontrakte, Objektorientierter Entwurf, UML-Notation, Objektorientierte Frameworks, Klassenbibliotheken, Entwurfsmuster, Testen von Software, Programmierstil.

### **Algorithmen und Datenstrukturen 1** (2VO + 1UE; SSW, Blaschek)

*Ziele:* Einführung in das algorithmische Denken und Kennenlernen der wichtigsten Algorithmen mit statischen Datenstrukturen. Die Lehrveranstaltung ist unabhängig von einer bestimmten Programmiersprache.

*Inhalt:* Algorithmusbegriff, Spezifikation, Algorithmen mit Gedächtnis, Komplexität, Rekursion, Zufallszahlen, Sortieren, Exhaustion, Zeichenkettensuche, Klassifikation von Algorithmen.

### **Betriebssysteme, Praktikum Betriebssysteme** (2VO + 1PR; FIM, Mühlbacher - Sonntag)

*Ziele:* Vorstellung der Arbeitsweise und Prinzipien von Betriebssystemen in herstellerunabhängiger Weise. Im zugehörigen Praktikum werden wichtige Konzepte systemnaher Programmierung sowie die Grundlagen von TCP/IP vermittelt.

*Inhalt:* Grundlagen, Klassifikation, Speicherverwaltung, Virtual Memory, Prozesse, PCB, CPU-Scheduling, Parallelität, Mutual Exclusion, Semaphore, Deadlocks / Systemverklemmung, Event Handling, Platten, Dateien und Dateisysteme, TCP/IP und Netzwerke, Fallstudie Unix/Linux, Fallstudie Windows XP.

### **Wirtschaftsgrundlagen für Informatiker** (2VO; BIO, Reschitzegger)

*Ziele:* Vermittlung von betriebswirtschaftlichem Basiswissen für die Entwicklung, den Einsatz und das Management von IT-Anwendungssystemen, zur Führung IT-fokussierter Organisationseinheiten und als Grundlage für Unternehmensgründungen im IT-Bereich.

*Inhalt:* Unternehmensplanung, -gründung und -organisation (Betrieblicher Leistungsprozess, Geschäftsmodelle, Businessplan, Standortplanung, Aufbau- und Ablauforganisation), Personalwesen (Planung, Einsatz, Entwicklung, Entlohnung, Motivationstheorien, Lernen und Wissensmanagement, Gruppe, Konflikte, Macht und Führung), Marketing (Absatzplanung, Strategien der Markterschließung, „Marketing-Mix“), Rechnungs- und Finanzwesen (Einnahmen/Ausgaben-Rechnung, doppelte Buchführung, Bilanz, Gewinn & Verlustrechnung, Kostenrechnung, Kapitalbeschaffung und Finanzplanung, Controlling).

## 3. Semester

### **Analysis** (2VO + 2UE; RISC, Schicho)

*Ziele:* Verständnis der wichtigsten Grundlagen aus der Analysis. Fähigkeit zur Anwendung von Methoden aus der Analysis in wissenschaftlichen und technischen Problemstellungen.

*Inhalt:* Funktionen, Reelle Zahlen, Folgen und Reihen, Stetigkeit, Differentialrechnung, komplexe Funktionen, Optimierung, Integralrechnung; Potenzreihen und Potenzreihenentwicklung, Einfache Differentialgleichungen.

### **Berechenbarkeit und Komplexität** (2VO + 1UE; RISC, Winkler)

*Ziele:* Einführung in theoretische Aspekte von Algorithmen und Programmen (formale Sprachen, Komplexitätstheorie, Entscheidbarkeitstheorie).

*Inhalt:* Algorithmen (endliche Automaten, reguläre Sprachen, Random Access Machine, Turing-Maschine, rekursiv aufzählbare Sprachen, rekursive Funktionen); Komplexität von Algorithmen (Komplexitätsmaße, Beispiele); Entscheidbarkeit (Akzeptierungs- und Halteproblem, Unentscheidbarkeitshierarchie); Komplexität von Problemen (vollständige Probleme, P/NP-Problem).

### **Computer Architecture 1** (3VO + 1UE; CA, Strumpfen)

*Goals:* This course offers a study of the principles underlying the design process of hardware and software elements of computer architecture.

*Contents:* Digital design techniques for combinational and sequential circuits, including arithmetic circuits and memory structures, instruction set architecture and assembly programming, processor microarchitecture, memory system architecture including cache hierarchies and virtual memory, I/O and interrupts, and methods for timing and performance analysis.

### **Algorithmen und Datenstrukturen 2** (2VO + 1UE; PC, Ferscha)

*Ziele:* Vermittlung fortgeschrittener Algorithmen und Datenstrukturen, insbesondere dynamische Datenstrukturen und ihre Anwendungen.

*Inhalt:* Dynamische Datenstrukturen (Listen, Bäume, Graphen, Stacks, Queues, Mengen), Heaps, Skip Listen, Splay Trees, fortgeschrittene Sortieralgorithmen (Mergesort, Heapsort, Radixsort), balancierte Bäume, Hashing, Graphalgorithmen, geometrische Algorithmen, Multimedia-Algorithmen, parallele Algorithmen, verteilte Algorithmen, heuristische Algorithmen.

### **Systemnahe Programmierung** (2PR; FIM+CA)

*Ziele:* Einführung in Konzepte und Sprachen zur systemnahen Programmierung. Querschnitt durch die Abstraktionsschichten oberhalb der Rechnerarchitektur und unterhalb der Anwendungsprogrammierung.

*Inhalt:* Assembler, Ablaufmodell, Virtuelle Maschinen, C, C++. Fokus auf eine Einführung in C Programmierung.

### **Netzwerke und Verteilte Systeme** (2VO + 1UE; FIM, Hörmanseder)

*Ziele:* Vermittlung grundlegender Netzwerk-Konzepte anhand des ISO-7-Schichten-Modells, TCP/IP und Ethernet. Kennenlernen von wichtigen Algorithmen für verteilte Systeme.

*Inhalt:* ISO/OSI-7-Schichten-Modell, Sicherungsschicht (CRC-Polynome), Ethernet, Switching, Routing, TCP/IP, ausgewählte Applikationsprotokolle, grundlegende Algorithmen in verteilten Systemen (Synchronisation, Election, RPCs, ...).

### **Rechtsgrundlagen für Informatiker** (2VO; FIM, Sonntag)

*Ziele:* Vermittlung grundlegender materieller Rechtsvorschriften bezüglich Computerprogrammen sowie ein Auftreten im Internet.

*Inhalt:* Domainnamen (Namens-, Marken-, und Wettbewerbsrechtlicher Schutz), Urheber- und Datenbankrecht, Softwarepatente, Elektronische Signaturen und Rechnungen, Websites (Haftung, Informationspflichten), Datenschutz.

## 4. Semester

### **Formal Models** (2VO + 1UE; FMV, Biere)

*Goals:* To understand formal modelling and specification techniques for computer science systems.

*Contents:* Reactive systems, finite automata, petri nets, process algebra, temporal logic.

### **Statistik** (2VO + 2UE; CA, Beran - Pölz)

*Ziele:* Einführung in die Methodenlehre der Statistik mit Betonung der statistischen Modellerstellung und des statistischen Schließens.

*Inhalt:* Grundlagen der Wahrscheinlichkeitsrechnung, Theorie der Zufallsvariablen, Spezielle Verteilungen, Mehrdimensionale Zufallsvariable, Testverteilungen, Beschreibende Statistik: Ein- und zweidimensionale Merkmale, Schätzen von Parametern: Punktschätzung und Intervallschätzung, Testen von Hypothesen.

### **Computer Architecture 2** (2VO + 1UE; CA, Strumpfen)

*Goals:* Advanced study of the elements and principles of computer architecture.

*Contents:* The relationship between instruction set architecture and microarchitecture manifests itself in a large design space for general-purpose computers. This course examines the influence of technological advances on architectural structure and style by contrasting the milestones of computer architecture: from stack machines, CISC, RISC, superscalar, multi-threaded, dataflow, VLIW, vector, to graph reducers and symmetric multiprocessors.

### **Praktikum Digitale Schaltungstechnik** (2PR; RIIC, Bauernfeind)

*Ziele:* Umsetzen des in Technische Informatik 1 und 2 gelernten Stoffs in Laborversuchen. Handhabung einfacher Messgeräte und Softwarewerkzeuge der Elektronik (Multimeter, Oszilloskop, Logikanalysator, Signalgenerator, Simulationswerkzeuge).

*Inhalt:* Analoge Schaltungstechnik: Realisierung einfacher Anlogschaltungen mit Widerständen, Kondensatoren, Dioden und Transistoren, Messen von Spannung, Strom und Ladevorgängen am Kondensator, Simulation analoger Schaltungen in SPICE.

Digitale Schaltungstechnik: Methoden zum Entwurf und Simulation kombinatorischer und sequentieller Logikschaltungen, Realisierung und Analyse dieser Schaltungen auf Gatterebene, Realisierung digitaler Schaltungen in programmierbaren Logikbausteinen, Einsatz eines 8-Bit Mikrokontrollers (Atmel AVR), hardwarenahe Mikrokontrollerprogrammierung.

### **Praktikum aus Softwareentwicklung 2** (2PR; SSW, Prähofer und andere)

*Ziele:* Praktisches Arbeiten mit einer umfangreichen objektorientierten Klassenbibliothek (Java-Klassenbibliothek). Kennenlernen von Grundprinzipien und Techniken von objektorientierten Frameworks. Design komplexer Anwendungsprogramme.

*Inhalt:* Utility-Klassen, Threading, Reflection, graphische Benutzeroberflächen (Swing), Streaming, Netzwerkprogrammierung, Security, Remote Methods, Internationalisierung, JDBC, Servlets und JSP, Web Services.

### **Multimediasysteme** (2VO + 1UE; TK, Kotsis)

*Ziele:* Einführung in Standards und Formate multimedialer Daten. Bewerten von Komponenten, Architekturen und Trends von Multimedia-Systemen.

*Inhalt:* Medientypen (Text, Grafik/Bilder, Audio, Video), Codierungs- und Kompressionsverfahren (allgemeine Prinzipien und Standards), Multimedia-Formate (CD/DVD, QuickTime, SMIL), Architektur von Multimedia-Systemen, Grundlagen von Multimedia Streaming, Digitales und Interaktives Fernsehen.

### **Ethik und Gender Studies** (2KV; FIM, Renöckl)

*Ziele und Inhalt:* Klärung von Humanitätsvorstellungen und -Aufgaben. Folgerungen für Ziele, Formen und Kriterien des Einsatzes von Naturwissenschaft und Technik, Probleme und Gefahren der Technik aus ethischer Sicht, Verantwortung des Informatikers.

## 5. Semester

### **Software Engineering** (2VO + 1UE; SEA, Egyed - Grünbacher)

*Goals:* This lecture conveys the principles, processes, methods, and tools for the engineering of software systems. We also discuss the different roles that software engineers play during software development—especially during the early phases of the software development life cycle.

*Contents:* Life Cycle of Software Engineering, Modeling during Software Development, Unified Modeling Language (UML), Requirements Engineering, Software Architecture, Component-Based Software Engineering, Verification and Validation of Software, Software Reuse, Software Maintenance and Evolution as well as Project Management and Economic Aspects of Software Engineering.

### **Übersetzerbau** (2VO + 2UE; SSW, Mössenböck)

*Ziele:* Vermittlung des in der Praxis relevanten Rüstzeugs zum Bau von Compilern und Compiler-ähnlichen Werkzeugen.

*Inhalt:* Grundlagen der formalen Sprachen (reguläre und kontextfreie Sprachen, endliche Automaten, Kellerautomaten), Lexikalische Analyse, Syntaxanalyse (rekursiver Abstieg, LR(1), LALR(1)), Attributierte Grammatiken, Symboltabellen, Codeerzeugung, virtuelle Maschinen, Compilergeneratoren. In den Übungen wird ein vollständiger Compiler implementiert.

### **Informationssysteme 2** (2VO + 1UE; BIO, Retschitzegger)

*Ziele:* Aufbauend auf Informationssysteme 1 werden aktuelle Konzepte und Technologien von Informationssystemen sowie physische Datenbankaspekte behandelt und in Oracle praktisch angewendet.

*Inhalt:* Objektorientierung und Datenbanken (objektrelationale Datenbankkonzepte und Mapping-Middleware, gespeicherte Prozeduren und Trigger), XML-Technologien und Datenbanken (DTDs, XMLSchema, XPath, XMLAPIs, XSLT, XQuery, Speicherung von XML-Dokumenten), Transaktionsmechanismen (Serialisierbarkeit, Fehlersicherheit, sperrende/ nicht-sperrende Verfahren, erweiterte Transaktionsmodelle), Optimierungsmechanismen (Prinzipien der Optimierung, Anfrageoptimierung, algebraische Optimierung).

### **Artificial Intelligence** (2VO + 1UE; CP, Widmer)

*Goals:* Introduction to basic models and techniques in the field of Artificial Intelligence (AI).

*Contents:* Definitions of AI, history of AI, current state of the field; motivating scenario: autonomous intelligent "agents" problem solving as a search process (uninformed and heuristic search algorithms); heuristic search in game playing; knowledge representation and logical inference (propositional and first-order logic; Prolog); representing and reasoning about uncertainty in Bayesian networks; basic notions of machine learning: inductive concept learning and reinforcement learning; basic notions of computer perception.

*Labs:* The exercise track accompanies and complements the lecture series Artificial Intelligence. Students will improve their understanding of the material taught in that class by regularly solving example problems (on paper).

### **Bioinformatics** (2VO; BIO, Bodenhofer)

*Goals:* Knowledge of the most important bioinformatics tools and algorithms.

*Contents:* This course provides insights into basic bioinformatics algorithms including pairwise sequence comparison, multiple sequence alignment, construction of phylogenetic trees (genealogies of animals or viruses), classification of DNA or protein sequences, and microarray data analysis. All tools and algorithms discussed in this course represent the current state of the art. Basics of molecular biology and biotechnology are provided up to the minimum that is necessary to understand the idea behind the bioinformatics tools and algorithms discussed in this course.

## **Projektorganisation** (2KV; SEA, diverse LVA-Leiter)

*Ziele:* Vermittlung der theoretischen und praktischen Grundlagen der Unternehmens- und Projektorganisation sowie des IT-Projektmanagements.

*Inhalt:* Grundlagen (Begriffe, Projektarten, Risikobewertung, Einflussfaktoren), Institutionale Projektorganisation (Organisationsformen, Kompetenzteilung, Projektleiter, Mitarbeiterführung, Projektgruppe, Methoden und Instrumente), Funktionale Projektorganisation (Projektplanung, Spezifikation und Pflichtenheft, Vorgehens- und Prozessmodelle, Aufwandsschätzung, Termin-, Ablauf- und Ressourcenplanung, Projektkontrolle- und -steuerung, Berichts- und Dokumentationssystem, Qualitätsmanagement).

## **6. Semester**

### **Embedded and Pervasive Systems** (2KV; PC, Ferscha)

*Ziele:* Vermittlung von Technologien, Anwendungen und Trends beim Einsatz und bei der Entwicklung von Embedded Systems.

*Inhalt:* Formale Modelle (VHDL, Statecharts, SDL, Petrinetze), Entwicklungsumgebungen, Globale Zeit und Uhrensynchronisation, Echtzeit-Betriebssysteme, Simulation und Validierung, Hardware/Software-Codesign, Embedded System Hardware (ASICs, Mikrokontroller, DSPs, Logikbausteine, Smartcards), Drahtlose Kommunikation (IrDA, RFID, IEEE 802.11, Bluetooth), Sensoren (optische, akustische, biometrische), Aktuatoren, Non-Standard I/O-Geräte.

### **Computer Graphics** (2VO + 1UE; CG, Bimber)

*Goals:* This course teaches basic computer graphics techniques as well as general concepts of real-time and photo-realistic rendering. It will also provide a glance on modern approaches in computer graphics as well as an introduction to OpenGL and GPU programming.

*Contents:* Introduction to computer graphics, transformations and projections, raster algorithms, depth handling, local shading and illumination, texture-mapping (including bump-, reflection-, environment-, and shadow-mapping), global illumination (including raytracing and radiosity), graphics pipelines, scene graphs, curves and surfaces (including Bezier, Splines, and NURBS), basics of animation, human vision and display technology.

## **Informatik-Wahlfach**

*Siehe Broschüre mit den Lehrinhalten der Masterstudien*