

Modulhandbuch Bachelor

Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen : Erstfach Metalltechnik; Vertiefung IN

Prüfungsordnungsversion: 2008

gültig für das Studiensemester: Wintersemester 2013/14

Erstellt am: Mittwoch 27. November 2013
aus der POS Datenbank der TU Ilmenau

Herausgeber: Der Rektor der Technischen Universität Ilmenau

URN: urn:nbn:de:gbv:ilm1-mhba-3472

- Archivversion -

Modulhandbuch

LA BA Berufl. Schulen
LA Berufliche Schulen -
Erstfach Metalltechnik

Prüfungsordnungsversion:2008

Vertiefung:IN

Erstellt am:
Mittwoch 27 November 2013
aus der POS Datenbank der TU Ilmenau

Inhaltsverzeichnis

Name des Moduls/Fachs	1.FS	2.FS	3.FS	4.FS	5.FS	6.FS	7.FS	Abschluss	LP	Fachnr.
	VSP	VSP	VSP	VSP	VSP	VSP	VSP			
Grundlagen der Informatik								FP	14	
Grundlagen und Diskrete Strukturen			4 2 0					PL 90min	7	1026
Programmierparadigmen				2 2 0				PL 60min	4	5378
Softwaretechnik					2 1 0			PL 90min	3	5370
Technische Informatik für Lehramt								FP	12	
Technische Informatik 2				2 1 0				PL 90min	4	1407
Praktikum Technische Informatik						0 0 1		SL	1	5365
Theoretische Informatik für Lehramt								FP	12	
Algorithmen und Datenstrukturen				2 2 0				PL 90min	4	198
Automaten und Formale Sprachen					2 1 0			PL 90min	4	5353
Effiziente Algorithmen						2 1 0		PL 15min	4	5347
Praktische Informatik für Lehramt								FP	12	
Betriebssysteme					2 1 0			PL 90min	4	252
Computergrafik					3 1 0			PL 60min	4	5367
Datenbanksysteme für IN					2 2 0			PL 60min	4	5368
Telematik 1						2 1 0		PL 90min	4	1749
Hauptseminar Informatik								FP	4	
Hauptseminar Informatik						0 2 0		PL	4	6992

Modul: Grundlagen der Informatik

Modulnummer6991

Modulverantwortlich: Univ.-Prof. Dr. Martin Dietzfelbinger

Modulabschluss: Fachprüfung/Modulprüfung generiert

Lernergebnisse

Die Studierenden kennen und beherrschen die grundlegenden Begriffe, Definitionen, Schlussweisen, Methoden und Aussagen derjenigen Bereiche der diskreten Mathematik, die für das Verständnis der nachfolgenden Informatikveranstaltungen notwendig sind. Sie erwerben Basiswissen über Programmiersprachparadigmen, einschliesslich der zugrunde liegenden Denk- und Verarbeitungsmodelle. Sie können Programmiersprachen und deren Konzepte nach wesentlichen Paradigmen klassifizieren. Sie sind in der Lage, zu gegebenen Problemen geeignete Paradigmen kritisch auszuwählen. Sie können einfache Programme sowohl in funktionaler als auch objektorientierter Programmiersprache systematisch entwerfen und implementieren. Die Studierenden erwerben grundlegendes Wissen über Vorgehens- und Prozessmodelle der Softwareentwicklung sowie über deren Basiskonzepte. Sie erwerben ein kritisches Grundverständnis zur Qualitätssicherung und zum Management der Softwareentwicklung. Ihnen wird Entscheidungskompetenz hinsichtlich Prinzipien, Methoden und Werkzeuge des Softwareentwurfs vermittelt. Sie verstehen das grundlegende Zusammenwirken unterschiedlicher Softwareentwicklungsphasen, und sie erwerben die Fähigkeit zur entwicklungsbezogenen, effektiven und koordinierten Teamarbeit.

Vorraussetzungen für die Teilnahme

Algorithmen und Programmierung

Detailangaben zum Abschluss

Grundlagen und Diskrete Strukturen

Fachabschluss: Prüfungsleistung schriftlich 90 min Art der Notengebung: Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: Wintersemester

Fachnummer: 1026 Prüfungsnummer: 2400072

Fachverantwortlich: Prof. Dr. Matthias Kriesell

Leistungspunkte: 7 Workload (h): 210 Anteil Selbststudium (h): 142 SWS: 6.0
 Fakultät für Mathematik und Naturwissenschaften Fachgebiet: 2411

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							4	2	0												

Lernergebnisse / Kompetenzen

Die Studierenden kennen und beherrschen die grundlegenden Begriffe, Definitionen, Schlussweisen, Methoden und Aussagen.

Vorkenntnisse

Abitur

Inhalt

Elementare Aussagenlogik, Mengen, Relationen, Abbildungen, elementare Kombinatorik, Zählprobleme, diskrete Stochastik, modulare Arithmetik, elementare algebraische Strukturen, Graphen

Medienformen

Tafelvortrag

Literatur

C. Meinel und M. Mundhenk, Mathematische Grundlagen der Informatik, Teubner, 3te Auflage, 2006 A. Steger, Diskrete Strukturen, Band 1 und 2, Springer

Detailangaben zum Abschluss

werden bei Bedarf festgelegt

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung MA

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung MA

Programmierparadigmen

Fachabschluss: Prüfungsleistung schriftlich 60 min Art der Notengebung: Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: unbekannt

Fachnummer: 5378 Prüfungsnummer: 2200049

Fachverantwortlich: Univ.-Prof. Dr.-Ing. Günter Schäfer

Leistungspunkte: 4 Workload (h): 120 Anteil Selbststudium (h): 75 SWS: 4.0
 Fakultät für Informatik und Automatisierung Fachgebiet: 2253

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										2	2	0									

Lernergebnisse / Kompetenzen

Fachkompetenz: Die Studierenden erwerben Basiswissen über Programmiersprachparadigmen, einschliesslich der zugrunde liegenden Denk- und Verarbeitungsmodelle. Sie können Programmiersprachen und deren Konzepte nach wesentlichen Paradigmen klassifizieren. **Methodenkompetenz:** Die Studierenden sind in der Lage, zu gegebenen Problemen geeignete Paradigmen kritisch auszuwählen. Sie können einfache Programme sowohl in funktionaler (Haskell 98) als auch objektorientierter (Java, C++) Programmiersprache systematisch entwerfen und implementieren. **Systemkompetenz:** Die Studierenden verstehen verschiedene Programmiersprachkonzepte im Kontext unterschiedlicher Programmiersprachen. **Sozialkompetenz:** Die Studierenden können erarbeitete Lösungen einfacher Programmieraufgaben in der Gruppe analysieren und bewerten.

Vorkenntnisse

Algorithmen und Programmierung (1. Semester)

Inhalt

* Übersicht über Programmierparadigmen * Funktionales Paradigma (FP, Haskell 98) o Funktionsdefinitionmöglichkeiten (Guards, Muster, lokale Defs) o Reduktionsstrategien (applicative, normal, lazy order) o Rekursion und Induktion o Höhere Funktionen (map, filter, fold, ...) o Algebraische und polymorphe Datentypen, Typklassen o Einfache Beweise o Module o Monaden * Objektorientiertes Paradigma (OOP) o OO-Konzepte, Begriffe, Notationen o Typen, Klassen, Subtyping und Vererbung o Abstrakte Klassen, Interfaces o Objektkommunikation o Dynamisches Binden, Polymorphismus o Ausnahmebehandlung o Design by Contract, Wiederverwendung o Generische Klassen o Mehrfachvererbung

Medienformen

Handouts

Literatur

wird in der Vorlesung bekannt gegeben

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

Bachelor Informatik 2013

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Softwaretechnik

Fachabschluss: Prüfungsleistung schriftlich 90 min Art der Notengebung: Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: unbekannt

Fachnummer: 5370 Prüfungsnummer: 2200072

Fachverantwortlich: Univ.-Prof. Dr.-Ing. Armin Zimmermann

Leistungspunkte: 3 Workload (h): 90 Anteil Selbststudium (h): 56 SWS: 3.0
 Fakultät für Informatik und Automatisierung Fachgebiet: 2236

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
													2	1	0						

Lernergebnisse / Kompetenzen

Fachkompetenz: Die Studierenden erwerben grundlegendes Wissen über Vorgehens- und Prozessmodelle der Softwareentwicklung, sowie über deren Methodik und Basiskonzepte. Sie können größere Entwicklungsaufgaben strukturieren, Lösungsmuster erkennen und anwenden, und verstehen den Entwurf von der Anforderungsermittlung bis hin zur Implementierung.

Methodenkompetenz: Den Studierenden wird Entscheidungskompetenz hinsichtlich möglicher Prinzipien, Methoden und Werkzeuge des ingenieurmäßigen Softwareentwurfs vermittelt.

Systemkompetenz: Die Studierenden verstehen das grundlegende Zusammenwirken unterschiedlicher Softwareentwicklungsphasen; anwendungsorientierte Kompetenzen bezüglich Modellierungsfähigkeit und Systemdenken werden geschult.

Sozialkompetenz: Die Studierenden verfügen über Fähigkeiten zur entwicklungsbezogenen, effektiven Teamarbeit.

Vorkenntnisse

Algorithmen und Programmierung

Inhalt

In der Lehrveranstaltung werden grundlegende Methoden, Modelle und Vorgehensweisen der Softwaretechnik bzw. des Software Engineering erlernt. Vorrangig wird die objektorientierte Sichtweise betrachtet, und in den Übungen anhand praktischer Beispiele vertieft. Für Implementierungsbeispiele wird vor allem JAVA verwendet.

- Einführung
- Modellierungskonzepte
 - . Überblick Modellierung
 - . klassische Konzepte (funktional, datenorientiert, algorithmisch, zustandsorientiert)
 - . Grundlagen Objektorientierung
 - . Unified Modeling Language (UML)
- Analyse
 - . Anforderungsermittlung
 - . Glossar, Geschäftsprozesse, Use Cases, Akteure
 - . Objektorientierte Analyse und Systemmodellierung
 - . Dokumentation von Anforderungen, Pflichtenheft
- Entwurf
 - . Software-Architekturen
 - . Objektorientiertes Design
 - . Wiederverwendung (Design Patterns, Komponenten, Frameworks, Bibliotheken)

- Implementierung
 - . Konventionen und Werkzeuge
 - . Codegenerierung
 - . Testen
- Vorgehensmodelle
 - . Überblick, Wasserfall, Spiralmodell, V-Modell XT, RUP, XP
- Projektmanagement
 - . Projektplanung
 - . Projektdurchführung

Medienformen

Präsentationsfolien, alle Unterlagen im Web verfügbar

Literatur

Balzert: Lehrbuch der Software-Technik. Spektrum 2000 Brügge, Dutoit: Objektorientierte Softwaretechnik. Pearson 2004
 Sommerville: Software Engineering. Pearson 2007 Oestereich: Analyse und Design mit UML 2.1. Oldenbourg 2006 sowie
 ergänzende Literatur (Angabe auf den Webseiten und in der Vorlesung)

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

Bachelor Mathematik 2009

Bachelor Mathematik 2013

Bachelor Technische Kybernetik und Systemtheorie 2010

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Master Mathematik und Wirtschaftsmathematik 2008

Master Medientechnologie 2009

Master Medientechnologie 2013

Modul: Technische Informatik für Lehramt

Modulnummer7009

Modulverantwortlich: Univ.-Prof. Dr.-Ing. habil. Wolfgang Fengler

Modulabschluss: Fachprüfung/Modulprüfung generiert

Lernergebnisse

Die Studierenden verstehen detailliert Aufbau und Funktionsweise von Prozessoren und Rechnern. Sie sind in der Lage, ein Beschreibungsmittel für die Modellierung von Strukturen und Abläufen mit formalen Mitteln anzuwenden sowie einfache maschinennahe Programme zu entwerfen und zu analysieren sowie Schaltungen und einfache algorithmische Abläufe auf maschinennahem Niveau zu entwerfen und zu implementieren. Die Studierenden verstehen detailliert Aufbau und Funktionsweise von fortgeschrittenen Prozessoren und Rechnern sowie Entwicklungstendenzen der modernen Rechner- und Systemarchitektur. Sie sind in der Lage, Anwendungsbeispiele und Architekturvarianten zu entwickeln. Sie analysieren Leistungskennwerte von Rechnern und Rechnersystemen. Sie erwerben fundierte Kenntnisse und Fertigkeiten zum Entwurf digitaler Systeme und können entworfene Schalsysteme bzgl. Aufwand und Korrektheit beurteilen. In der Vorlesung "Neuroinformatik 1" lernen die Studierenden die Grundlagen der Neuroinformatik und der Künstlichen Neuronalen Netze als wesentliche Säule der "Computational Intelligence" kennen. Sie verstehen die grundsätzliche Herangehensweise und kennen die wesentlichen Lösungsansätze, Modellierungs- und Implementierungstechniken beim Einsatz von neuronalen und probabilistischen Methoden im Unterschied zu klassischen Methoden der Informationsverarbeitung und Mustererkennung. Die Studenten verstehen die Komplexität beim Entwurf von eingebetteten und komplexen Automatisierungssystemen. Sie kennen die Probleme und Verfahren beim Entwurf von Echtzeitsystemen und können diese an einfachen Aufgabenstellungen anwenden.

Vorraussetzungen für die Teilnahme

keine Voraussetzungen

Detailangaben zum Abschluss

Technische Informatik 2

Fachabschluss: Prüfungsleistung schriftlich 90 min Art der Notengebung: Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: unbekannt

Fachnummer: 1407 Prüfungsnummer: 2200034

Fachverantwortlich: Univ.-Prof. Dr.-Ing. habil. Wolfgang Fengler

Leistungspunkte: 4 Workload (h): 120 Anteil Selbststudium (h): 86 SWS: 3.0
 Fakultät für Informatik und Automatisierung Fachgebiet: 2231

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										2	1	0									

Lernergebnisse / Kompetenzen

Fachkompetenz: Die Studierenden verstehen detailliert Aufbau und Funktionsweise von Prozessoren und Rechnern. Die Studierenden verstehen Entwicklungstendenzen der Rechnerarchitektur.

Methodenkompetenz: Die Studierenden sind in der Lage, ein Beschreibungsmittel für die Modellierung von Strukturen und Abläufen mit formalen Mitteln anzuwenden. Die Studierenden entwerfen und analysieren einfache maschinennahe Programme.

Sozialkompetenz: Die Studierenden sind in der Lage, praktische Problemstellungen der Rechnerarchitektur in der Gruppe zu lösen.

Vorkenntnisse

Vorlesung und Übung 'Technische Informatik 1' oder vergleichbare Veranstaltung

Inhalt

- Begriff der Rechnerarchitektur - Architekturmodellierung mit Petrinetzen - Innenarchitektur von Prozessoren - Befehlssatzarchitektur und Assemblerprogramme - Außenarchitektur von Prozessoren - Aufbau und Funktion von Speicherbaugruppen - Aufbau und Funktion von Ein- und Ausgabebaugruppen - Fortgeschrittene Prinzipien bei Rechnerarchitekturen

Medienformen

Vorlesung: Folien (Beamer erforderlich), Arbeitsblätter (Online und Copyshop) Übung: Arbeitsblätter und Aufgabensammlung (Online und Copyshop) Allgemein: Webseite (Materialsammlung und weiterführende Infos) <http://www.tu-ilmenau.de/ra>

Literatur

Primär: Eigenes Material (Online und Copyshop) Sekundär: W. Fengler, I. Philippow: Entwurf Industrieller Mikrocomputer-Systeme. ISBN 3-446-16150-3, Hanser 1991. C. Märtin: Einführung in die Rechnerarchitektur - Prozessoren und Systeme. ISBN 3-446-22242-1, Hanser 2003. T. Flik: Mikroprozessortechnik. ISBN 3-540-22270-7, Springer 2005. Allgemein: Webseite <http://tu-ilmenau.de/ra> (dort auch gelegentlich aktualisierte Literaturhinweise und Online-Quellen).

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Biomedizinische Technik 2008

Bachelor Ingenieurinformatik 2008

Bachelor Wirtschaftsinformatik 2009

Bachelor Wirtschaftsinformatik 2010

Bachelor Wirtschaftsinformatik 2011

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Praktikum Technische Informatik

Fachabschluss: Studienleistung alternativ Art der Notengebung: Testat / Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: unbekannt

Fachnummer: 5365 Prüfungsnummer: 2200051

Fachverantwortlich: Univ.-Prof. Dr.-Ing. habil. Wolfgang Fengler

Leistungspunkte: 1 Workload (h): 30 Anteil Selbststudium (h): 19 SWS: 1.0
 Fakultät für Informatik und Automatisierung Fachgebiet: 2231

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
																0	0	1			

Lernergebnisse / Kompetenzen

Fachkompetenz: Die Studierenden verstehen Aufbau und Funktionsweise zu Strukturen und Funktionen von digitaler Hardware, Prozessoren, Rechnern. Die Studierenden verstehen Syntax und Semantik von Beschreibungsmitteln für die Modellierung von Strukturen und Abläufen mit formalen Mitteln und von Assemblerbefehlen. Die Studenten verstehen Methoden zum Entwurf digitaler, sequentieller Steuerungssysteme und von Steuerprogrammen. Die Studierenden verstehen Aufbau und Funktionsweise zu verschiedenen Typen Neuronaler Netzwerke.

Methodenkompetenz: Die Studierenden sind in der Lage, einfache digitale Schaltungen, einfache algorithmische Abläufe auf maschinennahem Niveau und die Ansteuerung von E/A-Baugruppen und digitale Steuerungen zu entwerfen und zu implementieren. Die Studierenden können ausgewählte, zur Informationsverarbeitung gehörige mathematische Operationen berechnen. Die Studierenden sind in der Lage, Beschreibungsmittel für die Modellierung und Verifikation von Strukturen und Abläufen mit formalen Mitteln anzuwenden und die Ergebnisse zu interpretieren und kritisch zu beurteilen. Die Studierenden können computergestützte Werkzeuge für Entwurf digitaler Schaltungen, Assemblerprogrammierung, Modellierung und Steuerungsentwurf verwenden. Die Studierenden sind in der Lage, die internen Vorgänge von Neuronalen Netzen zu verstehen und deren Parametrisierung für verschiedene Anwendungsfälle vorzunehmen.

Systemkompetenz: Die Studierenden besitzen ein grundlegendes Verständnis für die Zusammenhänge zwischen modellierten und realen Abläufen im Bereich der Funktion von digitalen Schaltungen, Prozessoren, Rechnern, industriellen Steuerungen und Neuronalen Netzen. Sie erkennen den Zusammenhang zwischen verschiedenen Beschreibungsniveaus anhand praktischer Anwendung. Sie sind in der Lage, vorhandenes Wissen in begrenzter Zeit erfolgreich zur Problemlösung anzuwenden.

Sozialkompetenz: Die Studierenden lösen einen Teil der Aufgaben in der Gruppe. Sie sind in der Lage, auf Kritiken und Lösungshinweise zu reagieren. Sie verstehen die Notwendigkeit einer sorgfältigen und ehrlichen Arbeitsweise.

Vorkenntnisse

- Vorlesung und Übung Rechnerorganisation oder vergleichbare Veranstaltungen
- Vorlesung und Übung Rechnerarchitekturen 1 oder vergleichbare Veranstaltungen
- Vorlesung und Übung Prozessinformatik oder vergleichbare Veranstaltungen
- Vorlesung und Übung Neuroinformatik oder vergleichbare Veranstaltungen

Inhalt

Durchführung von mehreren Praktikumsversuchen zu folgenden Themen: • Kombinatorische Grundsaltungen • Informationskodierung • Petrinetze: Modellierung und Simulation • Einfache Assemblerprogramme • Ansteuerung unterschiedlicher E/A-Baugruppen • Hardware-Realisierung sequentieller Schaltungen • PLD-Realisierung sequentieller Schaltungen • Entwurf einer Transportstreckensteuerung mit Petri-Netzen • Programmwurf für die Steuerung eines Industriemodells • Funktionsweise Neuronaler Netze (supervised und unsupervised)

Medienformen

• Versuchsanleitungen • Internetpräsenz • Praxisnahe Softwarewerkzeuge und Versuchsaufbauten • Java-Applets.

Literatur

• Praktikumsanleitungen (Download und Copyshop) • Materialsammlungen der unter "Vorkenntnisse" genannten Vorlesungen und Übungen (siehe dort), weitere Literaturhinweise finden sich dort. Allgemein: Der primäre Anlaufpunkt ist der Webauftritt!
<http://tu-ilmenau.de/ra>, .../iks, .../neurob, .../sp

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Modul: Theoretische Informatik für Lehramt

Modulnummer7011

Modulverantwortlich: Univ.-Prof. Dr. Martin Dietzfelbinger

Modulabschluss: Fachprüfung/Modulprüfung generiert

Lernergebnisse

Die Studierenden kennen die Grundprinzipien des Algorithmenentwurfs und der Korrektheitsanalyse und Zeitanalyse von Algorithmen. Sie kennen Verfahren zur Spezifikation von Datentypen sowie die wesentlichen Algorithmenparadigmen und die wichtigsten grundlegenden Datenstrukturen sowie die Art und Weise der Verwendung dieser Strukturen in grundlegenden Algorithmen (insbesondere Sortierverfahren). Sie kennen die grundlegenden Datenstrukturen und die Basisalgorithmen für diese Strukturen. Die Studierenden sind in der Lage, sämtliche behandelten Algorithmen an Beispieleingaben auszuführen. Sie können einfache Datentypen syntaktisch und semantisch spezifizieren (mathematisches Modell). Sie überblicken die Umsetzung von Algorithmen in Programme. Sie können Zeitanalysen von einfachen iterativen und rekursiven Algorithmen vornehmen und Algorithmen nach ihrer Laufzeit klassifizieren und ihre Nützlichkeit einschätzen. Sie können in einfachen Problemstellungen geeignete Datenstrukturen auswählen und Algorithmenparadigmen einsetzen, um neue Algorithmen und Programme zu erstellen. Die Studierenden kennen die grundlegenden Strategien zum Entwurf effizienter Algorithmen und können sie beim Entwurf neuer Algorithmen anwenden. Sie können gegebene Standardalgorithmen an spezielle Situationen anpassen (ohne die Analyse zu vernachlässigen). Sie kennen die grundlegenden Analysemethoden und Klassifikationsinstrumente und können Algorithmen bezüglich ihres Ressourcenbedarfs analysieren. Die Studierenden kennen die Grundzüge der Theorie der Formalen Sprachen und der Automaten. Sie sind in der Lage, die behandelten Algorithmen und Konstruktionsverfahren an Beispieleingaben auszuführen (Automaten-, Grammatiktransformationen). Sie können Nicht-Regularitätsbeweise und Nicht-Kontextfreiheitsbeweise an Beispielen durchführen. Sie können für vorgegebene Sprachen / Probleme Automaten und/oder Grammatiken konstruieren. Sie können Entscheidungsverfahren und Transformationsverfahren für Automaten und Grammatiken anwenden.

Vorraussetzungen für die Teilnahme

Mathematik-Grundlagen, Algorithmen und Programmierung

Detailangaben zum Abschluss

Algorithmen und Datenstrukturen

Fachabschluss: Prüfungsleistung schriftlich 90 min

Art der Notengebung: Gestufte Noten

Sprache: Deutsch

Pflichtkennz.: Pflichtfach

Turnus: Sommersemester

Fachnummer: 198

Prüfungsnummer: 2200056

Fachverantwortlich: Univ.-Prof. Dr. Martin Dietzfelbinger

Leistungspunkte: 4	Workload (h): 120	Anteil Selbststudium (h): 75	SWS: 4.0
Fakultät für Informatik und Automatisierung			Fachgebiet: 2242

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										2	2	0									

Lernergebnisse / Kompetenzen

Fachkompetenz: Die Studierenden kennen die Grundprinzipien des Algorithmenentwurfs und der Korrektheitsanalyse und Zeitanalyse von Algorithmen und Datenstrukturen. Die Studierenden kennen beispielhaft Verfahren für die Spezifikation von Datentypen. Sie kennen die O-Notation und ihre Regeln, sowie die Anwendung der O-Notation bei der Laufzeitanalyse. Die Studierenden kennen die grundlegenden Datenstrukturen „Array“, „Liste“, „Stack“, „Queue“, „gerichteter Baum“ und „Binärbaum“ mit ihren Implementierungsmöglichkeiten und können die zentralen Performanzparameter benennen und begründen. Sie kennen den Datentyp „binärer Suchbaum“ mit seinen Methoden für Einfügen und Suchen und den Datentyp „Mehrwegsuchbaum“. Die Studierenden kennen die Algorithmen für mindestens eine Variante von balancierten Binärbäumen und können sie an Beispielen durchführen. Die Studierenden kennen das Prinzip von einfachen Hashverfahren, verstehen die Funktionsweise und können das zu erwartende Verhalten für die verschiedenen Verfahren beschreiben. Sie kennen Konstruktionen einfacher randomisierter Hashklassen. Die Studierenden kennen die grundlegenden Sortieralgorithmen (Quicksort, Heapsort, Mergesort sowie Radixsort), können die Korrektheit der Verfahren begründen und ihre Laufzeit berechnen. Sie kennen die untere Schranke für vergleichsbasierte Sortierverfahren sowie den grundlegenden Datentyp „Priority Queue“ und seine Implementierung auf der Basis von binären Heaps. Die Studierenden kennen die Grundbegriffe der Graphentheorie, soweit sie algorithmisch relevant sind, und können mit ihnen umgehen. Sie kennen die wesentlichen Datenstrukturen für die Darstellung von Graphen und Digraphen mit den zugehörigen Methoden und Performanzparametern, sowie einfache Graphdurchmusterungsverfahren (Breitensuche, Tiefensuche). Sie kennen einen Algorithmus für die Ermittlung eines minimalen Spannbaums.

Methodenkompetenz: Die Studierenden sind in der Lage, die behandelten Algorithmen an Beispielen auszuführen. Sie können Berechnungsprobleme und einfache abstrakte Datentypen wie Stack, Queue, Liste, Priority Queue und Varianten syntaktisch und semantisch spezifizieren (mathematisches Modell). Sie können mit Hilfe der O-Notation Zeitanalysen von einfachen iterativen und rekursiven Algorithmen vornehmen, Algorithmen nach ihrer Laufzeit klassifizieren und ihre Nützlichkeit einschätzen. Sie können die Korrektheitsanalyse für die in der Lehrveranstaltung untersuchten Algorithmen vornehmen und sie auf Varianten anpassen. Sie können für Problemstellungen geeignete Datenstrukturen und Algorithmen auswählen und deren Eignung nachweisen. Sie können Algorithmenparadigmen einsetzen, um für mit den behandelten Bereichen verwandte Themen neue Algorithmen zu finden und die Korrektheit zu beweisen und die Laufzeit zu analysieren.

Vorkenntnisse

Algorithmen und Programmierung, Grundlagen und Diskrete Strukturen, Mathematik für Informatiker 1

Inhalt

Spezifikation von Berechnungsproblemen und von abstrakten Datentypen. Analyse von Algorithmen: Korrektheitsbeweise für iterative und rekursive Verfahren, Laufzeitbegriff, O-Notation, Laufzeitanalyse. Grundlegende Datenstrukturen (Listen, Stacks, Queues, Bäume). Binäre Suchbäume, Mehrwegsuchbäume, balancierte Suchbäume (AVL- und/oder Rot-Schwarz-

Bäume, B-Bäume). Einfache Hashverfahren, universelles Hashing. Sortierverfahren: Quicksort, Heapsort, Mergesort, Radixsort. Priority Queues mit der Implementierung als Binärheaps. Grundbegriffe der Graphentheorie (ungerichtete und gerichtete Graphen, Markierungen an Knoten und Kanten, Wege und Kreise, Bäume und Wälder, Zusammenhangskomponenten). Datenstrukturen für Graphen (Adjazenzmatrix, Kantenliste, Adjazenzlisten, Adjazenzarrays). Durchmustern von Graphen: Breitensuche, Tiefensuche, Zusammenhangskomponenten, Entdecken von Kreisen. Minimale Spann bäume. Begleitend: Methoden für die Analyse von Laufzeit und Korrektheit.

Medienformen

Projektion von Folien, Folien und Übungsblätter auf der Vorlesungswebseite.

Literatur

- * G. Saake, K.-U. Sattler, Algorithmen und Datenstrukturen, 4. Auflage, dpunkt, 2010.
- * R. Sedgewick, Algorithmen in C++, Pearson Studium, 2001.
- * T. Ottmann, P. Widmayer, Algorithmen und Datenstrukturen, Spektrum Akademischer Verlag, 4., überarb. Aufl. 2002
- * R. H. Güting, S. Dieker: Datenstrukturen und Algorithmen, Teubner, 2004
- * T. Cormen, C. Leiserson, R. Rivest, C. Stein, Introduction to Algorithms, Second Edition, MIT Press 2001. (Auch auf deutsch erhältlich.)
- * M. T. Goodrich, R. Tamassia, Data Structures and Algorithms in Java, 2. Auflage, Wiley, 2003.

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

Bachelor Ingenieurinformatik 2008

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Master Wirtschaftsinformatik 2009

Master Wirtschaftsinformatik 2011

Automaten und Formale Sprachen

Fachabschluss: Prüfungsleistung schriftlich 90 min Art der Notengebung: Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: Wintersemester

Fachnummer: 5353 Prüfungsnummer: 2200057

Fachverantwortlich: Univ.-Prof Dr. Dietrich Kuske

Leistungspunkte: 4 Workload (h): 120 Anteil Selbststudium (h): 86 SWS: 3.0
 Fakultät für Informatik und Automatisierung Fachgebiet: 2241

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
													2	1	0						

Lernergebnisse / Kompetenzen

Fachkompetenz: Die Studierenden kennen die Grundzüge der Theorie der Formalen Sprachen und der Automaten (siehe Inhaltsangabe).

Methodenkompetenz: Die Studierenden sind in der Lage, die behandelten Algorithmen und Konstruktionsverfahren an Beispieleingaben auszuführen (Automaten-, Grammatiktransformationen). Sie können Nicht-Regularitätsbeweise und Nicht-Kontextfreiheitsbeweise an Beispielen durchführen. Sie können für vorgegebene Sprachen / Probleme Automaten und/oder Grammatiken konstruieren. Sie können Entscheidungsverfahren und Transformationsverfahren für Automaten und Grammatiken anwenden.

Vorkenntnisse

Fach "Algorithmen und Datenstrukturen" und "Mathematik für Informatiker 1 und 2"

Inhalt

- Deterministische endliche Automaten
- reguläre Sprachen, lexikalische Analyse
- Nichtdeterministische endliche Automaten
- Reguläre Ausdrücke
- Äquivalenzbeweise
- Erkennen von Nichtregularität
- Minimierung endlicher Automaten
- Allgemeine Grammatiken
- Kontextfreie Grammatiken und kontextfreie Sprachen
- Normalformen, insbesondere Chomsky-Normalform
- Ableitungsbäume und Ableitungen
- Kellerautomaten, Äquivalenz
- Parsing

Medienformen

Vorlesung: Folien, Folienkopien (Online)
 Übung: Übungsblätter (Online)

Literatur

- Schöning "Theoretische Informatik kurzgefasst"
- Hopcroft, Motwani, Ullman "Einführung in die Automatentheorie, Formale Sprachen und Komplexität"
- Asteroth, Baier "Theoretische Informatik"
- Wegener "Theoretische Informatik"

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

Bachelor Mathematik 2009

Bachelor Technische Kybernetik und Systemtheorie 2010

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Master Mathematik und Wirtschaftsmathematik 2008

Effiziente Algorithmen

Fachabschluss: Prüfungsleistung mündlich 15 min Art der Notengebung: Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: Sommersemester

Fachnummer: 5347 Prüfungsnummer: 2200058

Fachverantwortlich: Univ.-Prof. Dr. rer. nat. habil. Manfred Kunde

Leistungspunkte: 4 Workload (h): 120 Anteil Selbststudium (h): 86 SWS: 3.0
 Fakultät für Informatik und Automatisierung Fachgebiet: 2241

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
																2	1	0			

Lernergebnisse / Kompetenzen

Diese Lehrveranstaltung wird nicht mehr angeboten.

Vorkenntnisse

"Algorithmen und Datenstrukturen", "Algorithmen und Programmierung", "Mathematik für Informatiker 1 und 2", "Grundlagen und Diskrete Strukturen"

Inhalt

1. Sortieren und Auswahlproblem 2. Verwaltung von Mengen - Union-Find-Datenstrukturen - Fibonacci-Heaps - Binomial Queues 3. Graphalgorithmen - All-Pairs-Shortest-Paths (Floyd) - Transitive Hülle (Warshall) - Single-Source-Shortest-Paths (Dijkstra) - Minimale Spannbäume (Kruskal, Prim, Maggs/Plotkin) 4. Flüsse in Netzwerken (mit Anwendungen) - Ford-Fulkerson-Algorithmus - Algorithmus von Dinic - Bipartites Matching 5. Arithmetische Algorithmen - Multiplikation ganzer Zahlen - Matrixmultiplikation Designmethoden: Greedy, Dynamische Programmierung, Divide-And-Conquer, Backtracking. Analysemethoden: Divide-and-Conquer-Rekurrenzen, amortisierte Analyse

Medienformen

Tafel, Folien

Literatur

wird in der Vorlesung angegeben

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

Bachelor Mathematik 2009

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Master Mathematik und Wirtschaftsmathematik 2008

Modul: Praktische Informatik für Lehramt

Modulnummer6995

Modulverantwortlich: Univ.-Prof. Dr.-Ing. habil. Winfried Kühnhauser

Modulabschluss: Fachprüfung/Modulprüfung generiert

Lernergebnisse

Fachkompetenz: Die Studierenden erwerben grundlegendes Wissen über Aufbau und Funktionsweise von Betriebssystemen, Datenbanksystemen, Netzen (insbesondere des Internet) sowie Computergrafiksystemen.

Methodenkompetenz: Die Studierenden verfügen über das Wissen, Entwurfsmethoden im Kontext dieser Systeme einzusetzen. Sie können die Auswirkungen bestimmter Entwurfsentscheidungen auf grundlegende Leistungseigenschaften und -kenngrößen einschätzen. **Systemkompetenz:** Die Studierenden verstehen das grundlegende Zusammenwirken der Komponenten der Systeme und können diese in neuen Zusammenhängen anwenden. **Sozialkompetenz:** Die Studierenden erarbeiten Lösungen zu Aufgaben aus den Bereichen der Vorlesungen in der Gruppe und vertiefen bei Behandlung dieser Themen das Verständnis.

Vorraussetzungen für die Teilnahme

siehe individuelle Fächerbeschreibungen

Detailangaben zum Abschluss

keine

Betriebssysteme

Fachabschluss: Prüfungsleistung schriftlich 90 min

Art der Notengebung: Gestufte Noten

Sprache: Deutsch

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 252

Prüfungsnummer: 2200059

Fachverantwortlich: Univ.-Prof. Dr.-Ing. habil. Winfried Kühnhauser

Leistungspunkte: 4	Workload (h): 120	Anteil Selbststudium (h): 86	SWS: 3.0
Fakultät für Informatik und Automatisierung			Fachgebiet: 2255

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
													2	1	0						

Lernergebnisse / Kompetenzen

Die Kursteilnehmer sollen Betriebssysteme als strukturierte Systeme aus Komponenten mit individuellen Aufgaben und hochgradig komplexen Beziehungen verstehen; sie erwerben die Fähigkeit, Betriebssysteme bezüglich ihrer Eignung und Leistungen in unterschiedlichen Anwendungsdomänen zu analysieren, zu bewerten und einzusetzen.

Vorkenntnisse

Rechnerorganisation, Rechnerarchitekturen 1, Programmierparadigmen, Kommunikationsmodelle, Algorithmen und Programmierung, Algorithmen und Datenstrukturen

Inhalt

Betriebssysteme bilden das Software-Fundament aller informationstechnischen Systeme. Ihre funktionalen und vor allem ihre nichtfunktionalen Eigenschaften wie Robustheit, Sicherheit oder Wirtschaftlichkeit üben einen massiven Einfluss auf sämtliche Softwaresysteme aus, die unter ihrer Kontrolle ablaufen.

Dieser Kurs vermittelt Wissen über die grundlegenden Aufgaben, Funktionen und Eigenschaften von Betriebssystemen. Er stellt ihre elementaren Abstraktionen und Paradigmen vor und erklärt Prinzipien, Algorithmen und Datenstrukturen, mit denen funktionale und nichtfunktionale Eigenschaften realisiert werden.

Medienformen

Vorlesung mit Projektor und Tafel, über Web-Plattform, Skript/Folien-Handouts, Bücher, Fachaufsätze, Übungsblätter, Diskussionsblätter

Literatur

Siehe Webseiten der Veranstaltung

Detailangaben zum Abschluss

keine

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

Bachelor Informatik 2013

Bachelor Ingenieurinformatik 2008

Bachelor Ingenieurinformatik 2013

Bachelor Mathematik 2009

Bachelor Mathematik 2013

Bachelor Wirtschaftsinformatik 2009

Bachelor Wirtschaftsinformatik 2010

Bachelor Wirtschaftsinformatik 2011

Bachelor Wirtschaftsinformatik 2013

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Master Mathematik und Wirtschaftsmathematik 2008

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung AM

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung WM

Computergrafik

Fachabschluss: Prüfungsleistung schriftlich 60 min Art der Notengebung: Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: Wintersemester

Fachnummer: 5367 Prüfungsnummer: 2200060

Fachverantwortlich: Univ.-Prof. Dr. sc. techn. Beat Brüderlin

Leistungspunkte: 4 Workload (h): 120 Anteil Selbststudium (h): 75 SWS: 4.0
 Fakultät für Informatik und Automatisierung Fachgebiet: 2252

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
													3	1	0						

Lernergebnisse / Kompetenzen

Vermitteln der Grundlagen der Computergrafik bestehend aus Lineare Algebra/homogene Vektorräumen, Physik des Lichts, Rasteroperationen, Bildsynthese, Bildverarbeitung und effiziente geometrische Algorithmen und Datenstrukturen. Die Vorlesung bildet die Grundlagen für "photorealistische" Bildsynthese, wie sie in der Industrie sowie bei den Medien Verwendung finden (z. B. Filmindustrie, Computer-Aided Design, Computerspiele, Styling). Vermittlung von Grundlagen für weiterführende Vorlesungen: Geometrisches Modellieren, Interaktive Grafische Systeme / Virtuelle Realität, Technisch-wissenschaftliche Visualisierung, Fortgeschrittene Bildsynthese, Bildverarbeitung I & II.

Vorkenntnisse

Programmierkenntnisse Grundlagen Algorithmen & Datenstrukturen

Inhalt

Einführung: Überblick über das Fach Grafische Datenverarbeitung. Einführung: Vektoren und Matrizen, Transformationen, Homogene Vektorräume, 2D, 3D-Primitiven und Operationen, View-Transformationen Farbwahrnehmung, Tristimulus Ansatz, Farbmodelle: RGB, CMY, HSV, CIE. Spektrale Ansätze. Additive und Subtraktive Mischung. Lichtquellen und Filter. Rastergrafik-Hardware: Farbdiskretisierung, Farbbildröhre, LCD, Laserprinter, Ink-jet, etc. Rastergrafik: Rasterkonvertierung von Linien und Polygonen (Bresenham-Algorithmus, Polygonfüll-Algorithmus). Bildbearbeitung und Erkennung: Operationen auf dem Bildraster, Bildtransformationen (Skalierung, Drehung), Resampling und Filterung (Bilinear, Gauß) Dithering, Antialiasing, Flood Filling, Kantenverstärkung (Kantenerkennung) Licht und Beleuchtung: (physikalische Größen: Wellenlänge, Leuchtdichte, Leuchtstärke), Wechselwirkung von Licht und Material, Lichtausbreitung und Reflexion, Refraktion, Beleuchtungsmodelle, Materialeigenschaften (geometrische Verteilung) Farbige Lichtquellen (spektrale Verteilung) (Phong: diffuse, spekulare Reflexion). Cook-Torrance, Mehrfachreflexion, Lichteffekte: Schatten, Halbschatten, Kaustik. Bildsynthese: Rendering basierend auf Rasterkonvertierung: Z-Buffer, Flat-Shading, Gouraud shading, Phong Shading Global Illumination, Raytracing, Photontracing, Radiosity Texturemapping / Image-based Rendering: Affines und perspektivisches Texturemapping, projektives Texturemapping, Environment Mapping, Bumpmaps Effiziente Datenstrukturen zum räumlichen Sortieren und Suchen. Kd-Tree, Hüllkörper-Hierarchie, Anwendungen in der Grafik Ray-tracing, Kollisionserkennung. OpenGL, GPU-Renderpipeline, Szenegraphen, Effizientes Rendering grosser Szenen. Ausblick: Überblick geometrischer und physikalischer Modelldatenstrukturen: CSG, B-Rep, Voxel, Octree, parametrische Flächen Computergrafische Animation: (Key frame, motion curve, physikalisch basiertes Modellieren, Kollisionserkennung, Molekülmodelle)

Medienformen

Tafel, Folien, Buch Brüderlin, Meier: Computergrafik und geometrisches Modellieren (s. unten)

Literatur

Brüderlin, B., Meier, A., Computergrafik und geometrisches Modellieren, Teubner-Verlag, 2001 Weiterführende Literatur: José Encarnaçã, Wolfgang Straßer, Reinhard Klein: Graphische Datenverarbeitung 1: Gerätetechnik, Programmierung und Anwendung graphischer Systeme. 4th, revised and extended edition, Oldenbourg, Munich, Germany, 1996. José Encarnaçã, Wolfgang Straßer, Reinhard Klein: Graphische Datenverarbeitung 2: Modellierung komplexer Objekte und photorealistische Bilderzeugung. 4th, revised and extended edition, Oldenbourg, Munich, Germany, 1997. James D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes: Computer Graphics: Principles and Practice, Second Edition in C. - 2nd edition, Addison-Wesley, Reading, MA, USA, 1990. Alan Watt: 3D-Computergrafik. 3rd edition, Addison-Wesley, Reading, MA, USA, 2001.

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

Bachelor Informatik 2013

Bachelor Ingenieurinformatik 2008

Bachelor Mathematik 2009

Bachelor Mathematik 2013

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Master Mathematik und Wirtschaftsmathematik 2008

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung AM

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung WM

Datenbanksysteme für IN

Fachabschluss: Prüfungsleistung schriftlich 60 min Art der Notengebung: Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: unbekannt

Fachnummer: 5368 Prüfungsnummer: 2200061

Fachverantwortlich: Univ.-Prof. Dr.-Ing. habil. Kai-Uwe Sattler

Leistungspunkte: 4 Workload (h): 120 Anteil Selbststudium (h): 75 SWS: 4.0
 Fakultät für Informatik und Automatisierung Fachgebiet: 2254

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
													2	2	0						

Lernergebnisse / Kompetenzen

Nach dem Besuch dieser Veranstaltung können die Studierenden Datenbanksysteme anwenden. Sie kennen die Schritte des Entwurfs von Datenbanken und können die relationale Entwurfstheorie beschreiben. Weiterhin können sie deklarative Anfragen in SQL und XPath/XQuery formulieren sowie Integritätsbedingungen definieren.

Die Studierenden sind in der Lage, gegebene praktische Problemstellungen zu analysieren, im ER-Modell zu modellieren und in einer relationalen Datenbank abzubilden sowie SQL zur Anfrageformulierung zu nutzen.

Vorkenntnisse

Vorlesung Algorithmen und Programmierung

Inhalt

Grundbegriffe von Datenbanksystemen; Phasen des Datenbankentwurfs, Datenbankentwurf im Entity-Relationship-Modell, Relationaler Datenbankentwurf, Entwurfstheorie, Funktionale Abhängigkeiten und Normalformen; Grundlagen von Anfragen: Algebra und Kalküle; SQL: relationaler Kern und Erweiterungen, rekursive Anfragen mit SQL; Transaktionen und Integritätssicherung; Sichten und Zugriffskontrolle; XPath & XQuery als Anfragesprachen für XML

Medienformen

Vorlesung mit Präsentation und Tafel, Handouts, Moodle

Literatur

Saake, Sattler, Heuer: Datenbanken – Konzepte und Sprachen, 4. Auflage, mitp-Verlag, 2010m, 2002

Detailangaben zum Abschluss

keine

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

Bachelor Mathematik 2009

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Master Mathematik und Wirtschaftsmathematik 2008

Telematik 1

Fachabschluss: Prüfungsleistung schriftlich 90 min Art der Notengebung: Gestufte Noten
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: unbekannt

Fachnummer: 1749 Prüfungsnummer: 2200062

Fachverantwortlich: Univ.-Prof. Dr.-Ing. Günter Schäfer

Leistungspunkte: 4 Workload (h): 120 Anteil Selbststudium (h): 86 SWS: 3.0
 Fakultät für Informatik und Automatisierung Fachgebiet: 2253

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
																2	1	0			

Lernergebnisse / Kompetenzen

- **Fachkompetenz:** Die Studierenden verfügen über Kenntnisse und Überblickswissen zu Aufbau und Funktionsweise von Netzen, insbesondere des Internet.
- **Methodenkompetenz:** Die Studierenden sind in der Lage, einfache Protokollfunktionen zu spezifizieren und in Programmfragmente umzusetzen. Sie können die Auswirkungen bestimmter Entwurfsentscheidungen bei der Realisierung einzelner Protokollfunktionen auf grundlegende Leistungskenngrößen einschätzen. Sie kennen Darstellung von Protokollabläufen in Form von Message Sequence Charts und können gültige Protokollabläufe auf der Grundlage von Zustandsautomaten nachvollziehen.
- **Systemkompetenz:** Die Studierenden verstehen das grundsätzliche Zusammenwirken der Komponenten eines Netzes als System.
- **Sozialkompetenz:** Die Studierenden erarbeiten Problemlösungen einfacher Protokollfunktionen (z.B. Routing, Fehlerkontrolle, Flusskontrolle etc.) durch Bearbeiten von Übungsaufgaben in Gruppen und vertiefen bei Behandlung des Themas Geteilter Medienzugriff die technische Motivation für die Vorteile einer koordinierten Zusammenarbeit.

Vorkenntnisse

Hochschulzulassung;
 Grundlagenvorlesung in Informatik oder Programmierung (z.B. „Algorithmen und Programmierung“ oder eine vergleichbare Grundlagenvorlesung)

Inhalt

1. Einführung und Überblick: Grundsätzlicher Netzaufbau; Protokollfunktionen; Spezifikation; Architektur; Standardisierung; OSI- und Internet-Architekturmodell
2. Physikalische Schicht: Begriffe: Information, Daten und Signale; Physikalische Eigenschaften von Übertragungskanälen (Dämpfung, Verzerrung, Rauschen); Grenzen erreichbarer Datenübertragungsraten (Nyquist, Shannon); Taktsynchronisation; Modulationsverfahren (Amplituden-, Frequenz- und Phasenmodulation, kombinierte Verfahren)
3. Sicherungsschicht: Rahmensynchronisation; Fehlererkennung (Parität, Checksummen, Cyclic Redundancy Code); Fehlerbehebung (Forward Error Correction, Automatic Repeat Request); ARQ-Protokolle: Stop and Wait, Go-Back-N, Selective Reject; Medienzugriffsverfahren (ALOHA, Slotted ALOHA, Token-Ring, CSMA/CD); Ethernet; Internetworking: Repeater, Brücken und Router
4. Netzwerkschicht: Virtuelle Verbindungen vs. Datagramnetze; Aufgaben, Funktion und Aufbau eines Routers; Internet Protocol (IP): Paketaufbau und Protokollfunktionen, Hilfsprotokolle und Protokollversionen; Routingalgorithmen: Distanzvektor- und Link-State-Verfahren; Routingprotokolle des Internet (RIP, OSPF, BGP)
5. Transportschicht: Adressierung und Multiplexing; Verbindungsloser vs. verbindungsorientierter Transportdienst; Fehlerkontrolle; Flusskontrolle; Staukontrolle; Transportprotokolle des Internet (TCP, UDP)

Medienformen

Vorlesung mit Tafel und Folien-Präsentationen, Arbeitsblätter, Lehrbuch

Literatur

· A. S. Tanenbaum. Computernetzwerke. Pearson Education. · J. F. Kurose, K. W. Ross. Computernetze. Pearson Education.

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

Bachelor Ingenieurinformatik 2008

Bachelor Mathematik 2009

Bachelor Mathematik 2013

Bachelor Wirtschaftsinformatik 2009

Bachelor Wirtschaftsinformatik 2010

Bachelor Wirtschaftsinformatik 2011

Bachelor Wirtschaftsinformatik 2013

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Master Mathematik und Wirtschaftsmathematik 2008

Modul: Hauptseminar Informatik

Modulnummer5356

Modulverantwortlich: Univ.-Prof. Dr. Martin Dietzfelbinger

Modulabschluss: Fachprüfung/Modulprüfung generiert

Lernergebnisse

Ein Hauptseminar dient dem Einüben des Umgangs mit wissenschaftlichen und/oder anspruchsvollen technischen Texten in rezipierender und darstellender Weise. Hauptaufgabe eines Studierenden ist also die eigenständige, aber betreute Erarbeitung eines Stückes oder mehrerer Stücke fremder wissenschaftlicher Literatur bis zum eigenen Verständnis und die geschlossene Darstellung dieses Materials in einem Vortrag vor anderen Studierenden und dem Veranstalter, mit Befragung und Diskussion auf wissenschaftlichem Niveau. Eine schriftliche Zusammenfassung wird gefordert. Das Hauptseminar dient auch dazu, die Kommunikationsfähigkeit der Studierenden in einem fachlichen Kontext zu trainieren und zu bewerten. Für die Bewertung sind also der erzielte Grad von Verständnis des Stoffes, die Selbständigkeit der Vorbereitung, und besonders die Qualität des Vortrages in fachlicher und in gestalterischer Hinsicht heranzuziehen.

Vorraussetzungen für die Teilnahme

Grundlagen- und Wahlpflichtfächer des Studienplans für das Zweitfach Informatik, Semester 3-5

Detailangaben zum Abschluss

Hauptseminar Informatik

Fachabschluss: Prüfungsleistung alternativ
 Sprache: Deutsch

Art der Notengebung: Gestufte Noten
 Pflichtkennz.: Pflichtfach Turnus: ganzjährig

Fachnummer: 6992

Prüfungsnummer: 2200232

Fachverantwortlich: Univ.-Prof. Dr. Martin Dietzfelbinger

Leistungspunkte: 4	Workload (h): 120	Anteil Selbststudium (h): 98	SWS: 2.0
Fakultät für Informatik und Automatisierung			Fachgebiet: 2242

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
																0	2	0			

Lernergebnisse / Kompetenzen

Das Hauptseminar dient dem Einüben des Umgangs mit wissenschaftlichen und/oder anspruchsvollen technischen Texten in rezipierender und darstellender Weise. Hauptaufgabe eines Studierenden ist also die eigenständige, aber betreute Erarbeitung eines Stückes oder mehrerer Stücke fremder wissenschaftlicher Literatur bis zum eigenen Verständnis und die geschlossene Darstellung dieses Materials in einem Vortrag vor anderen Studierenden und dem Veranstalter, mit Befragung und Diskussion auf wissenschaftlichem Niveau (Fachkompetenz, Methodenkompetenz). Eine schriftliche Zusammenfassung wird gefordert. Das Hauptseminar dient auch dazu, die Kommunikationsfähigkeit der Studierenden in einem fachlichen Kontext zu trainieren und zu bewerten (Sozialkompetenz). Für die Bewertung sind also der erzielte Grad von Verständnis des Stoffes, die Selbstständigkeit der Vorbereitung und besonders die Qualität des Vortrages in fachlicher und in gestalterischer Hinsicht heranzuziehen.

Vorkenntnisse

Grundlagenstudium (1.-4. Semester)

Inhalt

Wechselndes Angebot aus den Informatik-Fachgebieten:

- Datenbanken und Informationssysteme
- Integrierte Hard- und Softwaresysteme
- Rechnerarchitekturen
- Softwaresysteme/Prozessinformatik
- Softwaretechnik und Programmiersprachen
- System- und Software-Engineering
- Telematik
- Verteilte Systeme und Betriebssysteme
- Graphische Datenverarbeitung
- Neuroinformatik und Kognitive Robotik
- Theoretische Informatik
- Komplexitätstheorie und Effiziente Algorithmen

Medienformen

Gestaltung der Vorträge mit Unterstützung moderner Medien.

Literatur

Originalliteratur. Wird von den jeweiligen Veranstaltern aus den einzelnen Fachgebieten bekanntgegeben.

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2010

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Elektrotechnik 2008 Vertiefung IN

LA BA Berufl. Schulen LA Berufliche Schulen - Erstfach Metalltechnik 2008 Vertiefung IN

Glossar und Abkürzungsverzeichnis:

LP	Leistungspunkte
SWS	Semesterwochenstunden
FS	Fachsemester
V S P	Angabe verteilt auf Vorlesungen, Seminare, Praktika
N.N.	Nomen nominandum, Nomen nescio, Platzhalter für eine noch unbekannte Person (wikipedia)
Objekttypen lt. Inhaltsverzeichnis	K=Kompetenzfeld; M=Modul; P,L,U= Fach (Prüfung,Lehrveranstaltung,Unit)