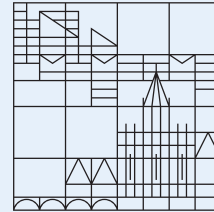


Universität
Konstanz

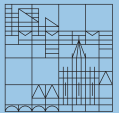


MODULHANDBUCH - MODULE DIRECTORY

Information Engineering Informatik

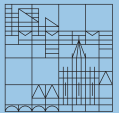
Fachbereich Informatik und Informationswissenschaft
Mathematisch-Naturwissenschaftliche Sektion
Bachelor / Master

Erstellungsdatum: 12. Juli 2013

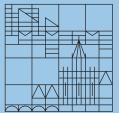


Inhaltsverzeichnis

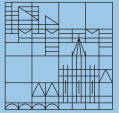
1	Information Engineering und Informatik	4
1.1	Abgrenzung	4
1.2	Aufbau des Studiums	5
1.3	Qualifikationsziele des Bachelorstudiengangs Information Engineering	5
1.4	Der Aufbau des Bachelorstudiengangs Information Engineering	6
1.5	Qualifikationsziele des Bachelorstudiengangs Informatik	7
1.6	Der Aufbau des Bachelorstudiengangs Informatik	8
1.7	Qualifikationsziele des Masterstudiengangs Information Engineering	10
1.8	Der Aufbau des Masterstudiengangs Information Engineering	10
2	Fachbereich Informatik und Informationswissenschaft	12
3	Musterstudienpläne	12
3.1	Bachelor Information Engineering	13
3.2	Master Information Engineering	14
3.3	Bachelor Informatik	15
4	Seminare und Projekte	16
4.1	Allgemeine Seminarbeschreibung	16
4.2	Allgemeine Bachelor-Projektbeschreibung	17
4.3	Allgemeine Master-Projektbeschreibung	17
5	Information Engineering and Informatics	18
5.1	Definition	18
5.2	Structure of the degree course	19
5.3	Qualification targets of the Information Engineering Bachelor Degree Course	19
5.4	Structure of the Information Engineering Bachelor degree course	20
5.5	Qualification targets of the Informatics Bachelor degree course	21
5.6	Structure of the Informatics Bachelor degree course	22
5.7	Qualification targets of the Information Engineering Master degree course	24
5.8	The structure of the Master’s degree course in Information Engineering	24
6	Department of Informatics and Information Science	25
7	Sample study plans	26
7.1	Bachelor in Information Engineering	27
7.2	Master in Information Engineering	28
7.3	Bachelor in Informatics	29
8	Seminars and Projects	30
8.1	Seminars	30
8.2	Bachelor Projects	31
8.3	Master Projects	31



9	Module Directory / Modulverzeichnis	32
10	Studiumsvorbereitung	32
10.1	Brückenkurs Mathematik	32
11	Grundstudium - 1. Semester - Pflichtbereich	34
11.1	Mathematische Grundlagen der Informatik (Mathematical Foundations of Computer Science)	34
11.2	Konzepte der Informatik (Principles of Computer Science 1)	35
11.3	Programmierkurs 1 (Programming Course 1)	37
11.4	Rechnersysteme und -netze (Computer Systems)	39
11.5	Analysis I	41
11.6	Schlüsselkompetenzen der Informatik (Key Competences for Undergraduate Students of Computer Science)	42
12	Grundstudium - 2. Semester - Pflichtbereich	44
12.1	Konzepte der Programmierung (Programming Concepts)	44
12.2	Programmierkurs 2 (Programming Course 2)	45
12.3	Datenbanksysteme (Database Systems)	46
12.4	Mathematik: Diskrete Strukturen (Mathematics: Discrete Structures)	48
13	Grundstudium - 3. Semester - Pflichtbereich	49
13.1	Betriebssysteme und systemnahe Programmierung	49
13.2	Programmierkurs 3 - Systemnahe Programmierung (Systems Programming)	50
13.3	Software Engineering (Software Engineering)	52
13.4	Algorithmen und Datenstrukturen (Algorithms and Data Structures)	53
13.5	Lineare Algebra I (Linear Algebra I)	54
13.6	Analyse & Visualisierung (Analysis & Visualization)	55
14	Grundstudium - 4. Semester - Pflichtbereich	57
14.1	Software-Projekt (Software Project)	57
14.2	Theoretische Grundlagen der Informatik (Theoretical Computer Science)	59
14.3	Computergrafik und interaktive Systeme (Computer Graphics and interactive Systems)	61
14.4	Statistik I	63
15	Vertiefungsstudium	64
15.1	Data Management in the Cloud (Data Management in the Cloud)	64
15.2	Geographic Information Systems (GIS) (Geographic Information Systems (GIS))	66
15.3	Computational Methods for Document Analysis (Computational Methods for Document Analysis)	68
15.4	Applied Visual Analytics (Applied Visual Analytics)	70
15.5	Neue Formen der Mensch-Computer-Interaktion: Blended Interaction (New Directions in Human-Computer Interaction: Blended Interaction)	72



15.6	Factorization Models for Machine Learning	74
15.7	Datenkompression (Data Compression)	75
15.8	Mustererkennung (Pattern Recognition)	77
15.9	Digital Libraries (Digital Libraries)	79
15.10	MindSwarms 2.0 - Swarm Networks	81
15.11	Data Mining: Foundations (Data Mining: Foundations)	82
15.12	Data Mining: Artificial Intelligence (Data Mining: Artificial Intelligence Methods)	83
15.13	Netzwerkanalyse (Network Analysis)	85
15.14	Zeichnen von Graphen (Graph Drawing)	86
15.15	Entwurf und Analyse von Algorithmen (Design and Analysis of Algorithms)	87
15.16	Modellierung in der Computergrafik (Modelling in Computer Graphics)	88
15.17	Methoden des Echtzeitrendering (Methods for Realtime Rendering)	89
15.18	Illustrative Computergrafik (Illustrative Computer Graphics)	90
15.19	3D-Computergraphik und Objektmodellierung (3D Computer Graphics and Object Modelling)	91
15.20	Multimedia database systems (Multimedia database systems)	93
15.21	Logic in Computer and Software Science	95
15.22	Model Checking of Software and Systems	96
15.23	Advanced Model Checking (Advanced Model Checking)	97
15.24	Data Warehousing und OLAP (Data Warehousing and OLAP)	98
15.25	Usability Engineering: Evaluation (Usability Engineering: Evaluation)	100
15.26	Visuelle Suchsysteme (Visual Information Seeking Systems)	102
15.27	Usability Engineering: Design (Usability Engineering: Design)	105
15.28	Camera Calibration for Computer Vision	107
15.29	Digitale Signalverarbeitung (Digital Signal Processing)	109
15.30	Architektur und Implementierung von Datenbanksystemen (Database System Architecture and Implementation)	111
15.31	Transaktionsorientierte Informationssysteme (Transactional Information Systems)	113
15.32	XML und Datenbanken (Database Supported XML)	115
15.33	Information Visualization I & II	116
15.34	Design Patterns and Concurrency (Design Patterns and Concurrency)	118
15.35	Directed Studies: Advanced Topics in Software and Systems Engineering	119
15.36	Introduction to Information Theory (Introduction to Information Theory)	120
15.37	Time Series Modeling (Time Series Modeling)	121



1 Information Engineering und Informatik

Da die Informatik nach vielen Jahren als Studienfach nicht mehr erklärungsbedürftig ist, wird in den folgenden Abschnitten nur das Information Engineering näher erläutert und beide Begriffe voneinander abgegrenzt, um die Unterschiede der Studienfächern aufzuzeigen.

1.1 Abgrenzung

Information Engineering umfasst für uns alle Aspekte der Gewinnung, Aufbereitung, Bereitstellung, Extraktion, Analyse und Vermittlung von Information. Wir verstehen es als eine angewandte Informatik, die sich mit allen entlang der Prozesskette

Daten - Information - Wissen

anfallenden Aufgaben beschäftigt. In ihrem Zentrum stehen daher Konzeption und Konstruktion informationsverarbeitender Systeme sowie entsprechende Methoden und Anwendungen. Beispiele für relevante Themen sind effiziente Codierungsverfahren zur Übermittlung von Multimedia-Daten, Anfrageauswertung in XML-Datenbanken, Data Mining in der Medikamentenherstellung oder die Visualisierung von Wetterdaten. Weiter gehören dazu auch die Nutzerfreundlichkeit von Computerprogrammen und Web-Designs sowie ethische Aspekte der Verfügbarkeit von Information. Das Fach vereint daher grundlegende Themen der praktischen und angewandten Informatik mit nutzungsorientierten Herangehensweisen der Informationswissenschaft.

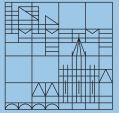
Die Ausbildung ruht (in dieser Reihenfolge) auf den drei Säulen:

1. Informatik
2. Umgang mit Information
3. Mathematik

mit einem Schwerpunkt im Bereich der Datenexploration und Visualisierung. Die Lehre ist nah an der Forschung des Fachbereichs - getreu dem Leitsatz der Universität Konstanz: „**Lehre aus Forschung**“. Der Fachbereich ist im Hinblick auf seinen Forschungsschwerpunkt „**Visualisierung und Exploration großer Datenräume**“ in Deutschland einzigartig breit ausgelegt und von hohem internationalen Ansehen.

Das Programm wird ergänzt durch einen von den Studierenden zu wählenden fachfremden Anteil, der darauf vorbereiten soll, dass die Arbeit unserer Absolventinnen und Absolventen in aller Regel den Umgang mit Informationsbedürfnissen aus anderen Anwendungsfeldern beinhaltet. Der durch Wirtschaft, öffentlichen Dienst und Gesellschaft entstehende Bedarf an der Beherrschung immer größer werdender Informationsmengen eröffnet hierbei ein weites Spektrum zur Umsetzung der erworbenen Kompetenzen. Typische Berufsbilder sind neben vielen informatiknahen Tätigkeiten vor allem im Umfeld der Informationsdienstleistungen zu finden.

Die klassische **Informatik** hat das Ziel, ein grundlagen- und anwendungsorientiertes Studium informationsverarbeitender Prozesse und Maschinen in ganzer Breite zu sein. Unser Studiengang vermittelt hierbei im Gegensatz zu Fachhochschulen und anderen Ausbildungsstätten insbesondere auch theoretische Konzepte und Methoden, die nicht aktuellen Trends unterliegen. Für die



Informatikerinnen und Informatiker im Berufsleben bedeutet dies eine weitgehende Unabhängigkeit von Branchentrends. Der Anteil an Mathematik und Theorie ist daher in diesem Fach höher. Da dieser Studiengang acht Semester dauert, gibt das Curriculum ausreichend Raum für eine Praxisphase (ein Semester Auslandsaufenthalt oder ein Praxissemester in der Wirtschaft, gerne auch im Ausland).

1.2 Aufbau des Studiums

Information Engineering wird seit dem Wintersemester 1999/2000 als erster konsekutiver Bachelor- und Masterstudiengang an der Universität Konstanz angeboten. Im Zuge der Akkreditierung durch die Fachagentur ASIIN im Jahre 2005 (die erste an der Universität Konstanz) wurde das Studienprogramm im Wintersemester 2006/2007 und im Wintersemester 2010/11 überarbeitet und weiter verbessert. Die nachfolgenden Erläuterungen sind eine Beschreibung der Studiengänge. Die **verbindlichen** Details sind in den gültigen **Prüfungsordnungen** der Bachelor- und Masterstudiengänge festgelegt.

Das Angebot beider Bachelorstudiengänge ist grundsätzlich deutschsprachig, für Vertiefungsveranstaltungen und die einschlägige Literatur sind in aller Regel aber auch Englischkenntnisse erforderlich.

Das Bachelorstudium Information Engineering ist auf drei Jahre angelegt, das Bachelorstudium Informatik auf vier Jahre. Beide vermitteln grundlegende Kenntnisse und Fähigkeiten, die zu typischen Berufen in der Informationsgesellschaft befähigen.

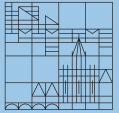
Direkt im Anschluss, nach einigen Jahren Berufserfahrung oder auch nach einem anderen Studium können vertiefte Kenntnisse im Rahmen des Masterstudiums erworben werden. Die Dauer des Masterstudiums hängt dabei vom Bachelorabschluss ab. Bei dreijährigen Bachelorabschlüssen dauert das Masterstudium zwei Jahre, bei vierjährigen äquivalenten Bachelorstudiengängen dauert das Masterstudium ein Jahr.

Jedes Studienjahr ist in zwei Hälften geteilt, das Winter- und das Sommersemester. Abgesehen von den Abschlussarbeiten sind alle Prüfungen studienbegleitend, was bedeutet, dass die Inhalte der Veranstaltungen jeweils bis zum Ende des Semesters, in dem das Modul angeboten wird, geprüft werden. Prüfungen finden in Form von Hausarbeiten, mündlichen Prüfungen oder Klausuren statt.

Der zu erwartende Aufwand für eine Veranstaltung wird gemäß ECTS (European Credit Transfer System) in Credits gemessen, wobei ein Credit für 30 Stunden Arbeitsaufwand steht. In beiden Studiengängen sind pro Semester 30 ECTS-Credits zu erwerben.

1.3 Qualifikationsziele des Bachelorstudiengangs Information Engineering

Der Studiengang „Information Engineering“ will einen „Informationsingenieur“ ausbilden, der in einer nutzungszentrierten Sichtweise in den großen Informationsräumen, die unsere Informationsgesellschaft charakterisieren, Daten sucht, filtert, organisiert, und als Information aufbereitet, zusammenfasst und präsentiert. Hierzu werden, aufbauend auf den Grundlagen der Informatik und Mathematik, Methoden, Anwendungen und Bewertungen von Systemen der automatischen Informationsverarbeitung vermittelt.



Das Bachelorstudium Information Engineering vermittelt Grundlagen, um aus einer nutzungsorientierten Perspektive Daten zu sammeln und aufzubereiten, um daraus Informationen zu extrahieren und zu präsentieren und das so gewonnene Wissen kompetent beurteilen und anwenden zu können. Dabei ist die Kombination von theoretischer und praktischer Informatik mit nutzungsorientierten Methoden der Informationswissenschaft das Studienziel für die Absolventen.

Bei der Entwicklung des Studienprogramms wurde besonders darauf geachtet, dass einerseits theoretisches Grundlagenwissen vermittelt wird, andererseits die Anwendungsorientierung im Sinne eines berufsqualifizierenden Studiums einen angemessenen Stellenwert erhält.

Kompetenzen. Besonderes Augenmerk wurde bei der Entwicklung des Studienprogramms auf die Kompetenzvermittlung im Einsatz von Informationssystemen unter technischen, administrativen und strategischen Aspekten in modernen Organisationen gelegt. Auch sollen die Absolventen in der Lage sein, in Institutionen der Informationswirtschaft zu arbeiten, die für die Bereitstellung von Informationsprodukten zuständig sind.

Absolventen verfügen über grundlegende wissenschaftliche Fachkenntnisse und überblicken die Zusammenhänge des Faches Information Engineering.

1.4 Der Aufbau des Bachelorstudiengangs Information Engineering

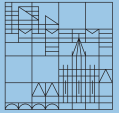
Der Bachelorstudiengang Information Engineering ist eine Kombination aus Grundlagen der Informationsverarbeitung und den im Fachbereich vorhandenen Vertiefungsmöglichkeiten in attraktiven Bereichen, vor allem der Exploration und Visualisierung von Information.

Den Zugang zum Studium regelt die Zulassungs- und Immatrikulationsordnung der Universität Konstanz. Da über die darin festgelegten Bedingungen hinaus keine Beschränkungen bestehen, reicht die Allgemeine Hochschulreife.

Das Bachelorstudium besteht aus vier Semestern Grundstudium und zwei Semestern Vertiefungsstudium. Folgende Pflichtveranstaltungen müssen im Grundstudium absolviert werden:

Informatik

- Modul Informatik 1 (Konzepte der Informatik und Programmierkurs 1)*
- Modul Informatik 2 (Konzepte der Programmierung und Programmierkurs 2)
- Rechnersysteme und -netze
- Algorithmen und Datenstrukturen
- Theoretische Informatik
- Software Engineering
- Softwareprojekt



Umgang mit Information

- Datenbanksysteme*
- Analyse und Visualisierung von Information
- Computergrafik und interaktive Systeme

Mathematik

- Mathematische Grundlagen der Informatik
- Diskrete Strukturen
- Statistik oder Numerik

Die mit * markierten Veranstaltungen müssen für das Bestehen der **Orientierungsprüfung** bis zum Ende des zweiten Semesters erfolgreich absolviert werden.

Für **fachfremde Module** existiert eine Vorschlagsliste von bewährten und sinnvollen Kombinationen. Es ist auch möglich, von zentraler Stelle angebotene Veranstaltungen für Schlüsselqualifikationen zu belegen.

Das **Vertiefungsstudium** ermöglicht die Wahl von Veranstaltungen aus den verschiedenen Forschungsgebieten des Information Engineering und ist stark projektorientiert. Dabei stellt das Bachelorprojekt den Fokus des Vertiefungsstudiums dar. Im letzten Semester wird über einen Zeitraum von vier Monaten eine wissenschaftliche Abschlussarbeit, die Bachelorarbeit, angefertigt, deren Inhalte in Form eines Kolloquiums auch mündlich geprüft werden.

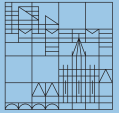
Erfolgreichen Absolventinnen und Absolventen wird der akademische Grad eines **Bachelor of Science (B.Sc.) im Fach Information Engineering** verliehen. Sie sind für vielfältige Aufgaben in der Informationsgesellschaft gewappnet und befähigt, in das Berufsleben einzusteigen. Sie können ihre Kenntnisse und Fähigkeiten aber auch im Rahmen des Masterstudiums Information Engineering oder eines anderen verwandten Masterstudiengangs weiter ausbauen.

1.5 Qualifikationsziele des Bachelorstudiengangs Informatik

Ziel des Informatik-Bachelorstudiengangs ist die Vermittlung der grundlegenden Methoden der angewandten Informatik mit einer soliden theoretischen und mathematischen Kompetenz.

Die Informatik nimmt in einer beständig zunehmenden Zahl von Branchen eine Schlüsselrolle ein. Die hohe Relevanz des Gebietes kann daran abgelesen werden, dass führende Unternehmen aller Branchen den Informatikbereich als einen strategischen Schwerpunkt identifiziert haben. Das Portfolio des Bachelorstudiengangs Informatik greift den Expertisebedarf potentieller Arbeitgeber direkt auf.

Wesentliche Zielgruppe sind Schulabgänger, die einen grundständigen Informatik-Studiengang aufnehmen wollen und von Vertiefungsmöglichkeiten im Forschungsschwerpunkt des Fachbereichs Informatik und Informationswissenschaft, einer curricular verankerten externen Praxisphase im Ausland oder in der Wirtschaft profitieren wollen. Ferner soll ein Direkteinstieg in ein



zukunftsträchtiges Arbeitsmarktsegment oder ein weiterführendes Studium bzw. eine Promotion möglich sein.

Diese Studienanfänger unterscheiden sich von denen, die das Bachelorstudium Information Engineering aufnehmen, durch größeres Interesse an mathematischen und systemnahen Grundlagen.

Der Studiengang ist

1. berufsqualifizierend (statt -befähigend),
2. vierjährig mit einsemestriger externer Praxisphase,
3. mit Fast Track-Option zur anschließenden Aufnahme einer Promotion,
4. mit Mentorenprogramm für eine leichtere Studierbarkeit und
5. mit hoher Durchlässigkeit von und zu dem bestehenden Bachelorstudiengang Information Engineering und dem modularisierten Lehramtshauptfach Informatik konzipiert.

Kompetenzen. Absolventinnen und Absolventen der Informatik sind in der Lage, mit wachsenden Datenmengen in verarbeitenden Systemen umzugehen und neben der anerkannten Fähigkeit zur Herstellung und Qualitätsprüfung neuer Software insbesondere die methodische Kompetenz zur Exploration und Visualisierung von Informationsräumen als Basis für die Entscheidungsfindung zu besitzen.

Das wissenschaftliche Ausbildungsprofil der Absolventen ist daher nicht nur für die akademische Forschung, sondern insbesondere auch für die Nachfrage der Wirtschaft konzipiert. Dieser Bedarf besteht unter anderem in der Forschung und Entwicklung der produzierenden Industrie und des Informationsmarktes.

1.6 Der Aufbau des Bachelorstudiengangs Informatik

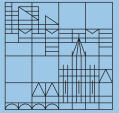
Das Bachelorstudium Informatik setzt auf eine breitere Basis in den Bereichen Mathematik und Informatikgrundlagen. Zusätzlich ermöglichen die acht Semester die Wiedereinführung eines Praxis- oder Auslandssemesters, für das 30 ECTS in der Prüfungsordnung vorgesehen sind. Im Vertiefungsstudium werden zusätzlich 30 ECTS in den Forschungsgebieten des Fachbereichs Informatik und Informationswissenschaft absolviert.

Den Zugang zum Studium regelt die Zulassungs- und Immatrikulationsordnung der Universität Konstanz. Da über die darin festgelegten Bedingungen hinaus keine Beschränkungen bestehen, reicht die Allgemeine Hochschulreife.

Das Bachelorstudium besteht aus vier Semestern Grundstudium und vier Semestern Vertiefungsstudium. Folgende Pflichtveranstaltungen müssen im Grundstudium absolviert werden:

Informatik

- Modul Informatik 1 (Konzepte der Informatik und Programmierkurs 1)*
- Modul Informatik 2 (Konzepte der Programmierung und Programmierkurs 2)



- Modul Informatik 3 (Betriebssysteme und Systemnahe Programmierung und Programmierkurs 3)
- Rechnersysteme und -netze
- Algorithmen und Datenstrukturen
- Theoretische Informatik
- Software Engineering
- Softwareprojekt

Umgang mit Information

- Datenbanksysteme*
- Analyse und Visualisierung von Information
- Computergrafik und interaktive Systeme

Mathematik

- Analysis 1
- Lineare Algebra 1
- Diskrete Strukturen
- Statistik oder Numerik

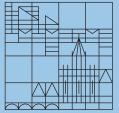
Die mit * markierten Veranstaltungen müssen auch hier für das Bestehen der **Orientierungsprüfung** bis zum Ende des zweiten Semesters erfolgreich absolviert werden. Im Rahmen eines vierjährigen Bachelors muss zudem eine **Zwischenprüfung** (ebenso studienbegleitend, ohne zusätzlichen Prüfungsaufwand) mit dem Absolvieren des vierten Semesters bestanden werden.

Fachfremde **Module** werden im Vertiefungsstudium, Schlüsselqualifikationen im Grund- und Vertiefungsstudium belegt.

Das **Vertiefungsstudium** nähert beide Bachelorstudiengänge wieder aneinander an. Die Informatiker besuchen Veranstaltungen aus den verschiedenen Forschungsgebieten des Information Engineering im Wahlbereich. Das Bachelorprojekt kann während der Semester 5 bis 7 absolviert werden. Im achten Semester wird die Bachelorarbeit angefertigt, deren Inhalte in Form eines Kolloquiums wiederum mündlich geprüft werden.

Im Grundstudium und im Vertiefungsstudium werden hier Studienleistungen im Umfang von 120 ECTS-Credits absolviert, so dass insgesamt 240 ECTS-Credits an Studienaufwand geleistet werden.

Die Abschlussnote setzt sich (zu unterschiedlichen Anteilen) aus den studienbegleitend erworbenen Noten und denen der Bachelorarbeit und des Kolloquiums zusammen.



Erfolgreichen Absolventinnen und Absolventen wird der akademische Grad eines **Bachelor of Science (B.Sc.) im Fach Informatik (Computer Science)** verliehen. Die Absolventen sind für die breite Vielfalt an Aufgaben in der globalen Informationsgesellschaft befähigt. Der vierjährige Abschluss ist berufsqualifizierend und bringt durch das Praxissemester bereits erste Berufserfahrungen mit sich. Für die Aufnahme eines Masterstudiums bedeutet der vierjährige Abschluss eine **Verkürzung des Masterstudiengangs** auf ein Jahr. Ebenso wird hier der **Direkteinstieg zur Promotion** möglich.

1.7 Qualifikationsziele des Masterstudiengangs Information Engineering

Der Studiengang „Information Engineering“ will einen „Informationsingenieur“ ausbilden, der in einer nutzungszentrierten Sichtweise in den großen Informationsräumen, die unsere Informationsgesellschaft charakterisieren, Daten sucht, filtert, organisiert und als Information aufbereitet, zusammenfasst und präsentiert. Hierzu werden, aufbauend auf den Grundlagen der Informatik und Mathematik, Methoden, Anwendungen und Bewertungen von Systemen der automatischen Informationsverarbeitung vermittelt.

Das Angebot des Masterstudiengangs wendet sich an internationale Absolventen der Bachelorstudiengänge Information Engineering und Informatik (Computer Science) sowie qualifizierte Quereinsteiger mit überdurchschnittlichem Hochschulabschluss in einem verwandten Fach.

Ziel des Studiengangs ist es, das grundlegende Wissen aus dem Bachelorstudiengang Information Engineering oder in Bachelorstudiengängen der Informatik (Computer Science) in verschiedene Richtungen zu vertiefen.

Die entsprechenden Studienmodule sollen die Studierenden befähigen, wissenschaftliche Methoden und Erkenntnisse bei informatisch schwierigen und komplexen Problemstellungen sowohl in der Praxis als auch in der Forschung einzusetzen. Von seiner inhaltlichen Ausrichtung gehört der Masterstudiengang zu den stärker forschungsorientierten Studiengängen. Die Lehre wird von Dozenten getragen, die vor allem aus Erfahrungen aktueller Forschung schöpfen. Die Studierenden werden frühzeitig in laufende Forschungsprojekte im Rahmen des vorgeschriebenen Praktikums eingebunden und vertiefen dieses Wissen später dann im Rahmen der Masterabschlussarbeit.

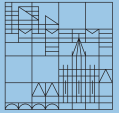
Kompetenzen. Durch die Masterprüfung erhalten Absolventen vertiefte wissenschaftliche Fachkenntnisse des Information Engineering und sind in der Lage, nach wissenschaftlichen Grundsätzen selbständig zu arbeiten und wissenschaftliche Methoden und Erkenntnisse anzuwenden.

Die Umstellung der Lehrsprache im Masterstudiengang Information Engineering auf Englisch bereitet auch die Masterstudierenden auf Karrieren in international operierenden Unternehmen, Projekten und weiterführende Studiengänge vor.

Absolventen sind Experten der nutzungorientierten Verarbeitung und Aufbereitung von Information.

1.8 Der Aufbau des Masterstudiengangs Information Engineering

Der Masterstudiengang Information Engineering vertieft die Kenntnisse des Bachelorstudiengangs und beinhaltet wieder Informatik entlang der Exploration von Daten, deren Visualisierung sowie



entsprechender Interaktionsformen mit dem Menschen.

Für die Zulassung zum Masterstudium gibt es eine Zulassungssatzung. Erfolgreiche Bewerber sind im Normalfall Absolventen einschlägiger Studiengänge (insbesondere Bachelor Information Engineering und Informatik) mit gutem Abschluss. Auch Quereinsteiger können zugelassen werden, wenn sie durch anrechenbare Vorleistungen oder im Rahmen einer Zulassungsprüfung die notwendigen Vorkenntnisse nachweisen.

Das Masterstudium besteht aus einem Wahlpflichtbereich und einem Wahlbereich, in dem auch die abschließende Masterarbeit angefertigt und im Rahmen eines Kolloquiums (mündliche Prüfung) präsentiert wird. In diesem **Wahlpflichtbereich** wird eines der Forschungsgebiete des Information Engineering als Schwerpunkt gewählt. Aus dem Schwerpunkt und zusätzlichen Vertiefungsveranstaltungen des **Wahlbereichs** kann das Studium individuell zusammengestellt werden. Da quer eingestiegene Studierende ein anderes Fach studiert haben, vertiefen sie sich im umfangreichen Wahlbereich des Information Engineering ausgiebiger, ohne die Option, auch fachfremde Lehrveranstaltungen (im Umfang von 17 ECTS-Credits) belegen zu können.

Die Dauer des Wahlbereichs hängt von der Zulassung für den einjährigen oder zweijährigen Master ab. Im Fall des einjährigen Masters wird mit dem Masterprojekt begonnen und parallel 17 ECTS-Credits vertiefende Veranstaltungen belegt. Im zweiten Semester wird hier bereits die Masterarbeit mit dem Kolloquium absolviert. Im Fall des zweijährigen Masters stehen für den Wahlpflicht- und den Wahlbereich drei Semester zur Verfügung. Hier werden dann 60 oder 77 ECTS-Credits an vertiefenden Lehrveranstaltungen belegt. Fachfremde Veranstaltungen sind für eine Gruppe (mit äquivalentem Bachelorabschluss) im Rahmen von 17 ECTS-Credits optional möglich.

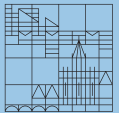
Kern der Studienplanung ist im ersten Semester ein obligatorisches Mentorengespräch, das den individuellen Studienplan zum Inhalt hat und auch mögliche Studienprofile definieren kann. Ein Studienprofil kann eine zusätzliche thematische Fokussierung bringen. Studienprofile sind optional und werden als Zusatz auf dem Zeugnis erwähnt.

Die Abschlussnote setzt sich (zu unterschiedlichen Anteilen) aus den studienbegleitend erworbenen Noten und denen der Masterarbeit und des Kolloquiums zusammen.

Erfolgreichen Absolventinnen und Absolventen wird der akademische Grad eines **Master of Science (M.Sc.) in Information Engineering** verliehen. Sie sind Experten für die Verarbeitung von Information und für weiterführende Aufgaben in der Informationsgesellschaft qualifiziert. Sie können ihren wissenschaftlichen Interessen auch im Rahmen einer Promotion in Informatik oder einem anderen verwandten Fach weiter nachgehen. Ein mögliches Studienprofil kann ergänzend als Schwerpunktkennzeichnung auf dem Zeugnis vermerkt werden.

Studienprofile sind **Modelle**, mit deren Hilfe sich die Masterstudentinnen und Masterstudenten einen individuellen Studienschwerpunkt bilden können. Sie geben dem Masterstudiengang Information Engineering eine Spezialisierung im weiten Feld der angewandten und praktischen Informatik:

- Visual Computing / Computervisualistik
- Mensch-Computer-Interaktion
- Digital Libraries
- Data Mining



- Information Systems Engineering

2 Fachbereich Informatik und Informationswissenschaft

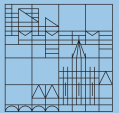
Mit seinen neun Professoren, vier Juniorprofessoren, zwei Hochschuldozenten und 80 Wissenschaftlerinnen und Wissenschaftlern deckt er das für einen Informatikstudiengang benötigte Fachspektrum ab und ist im Hinblick auf seinen Forschungsschwerpunkt **Exploration und Visualisierung großer Datenmengen einzigartig** und exzellent besetzt.

- Prof. Dr. Michael Berthold (Bioinformatik und Information Mining)
- Prof. Dr. Ulrik Brandes (Algorithmik)
- Prof. Dr. Oliver Deussen (Computergrafik und Medieninformatik)
- Prof. Dr. Daniel Keim (Datenbanken, Data Mining und Visualisierung)
- Prof. Dr. Stefan Leue (Software Engineering)
- Prof. Dr. Harald Reiterer (Mensch-Computer-Interaktion)
- Prof. Dr. Dietmar Saupe (Multimedia-Signalverarbeitung)
- Prof. Dr. Marc Scholl (Datenbanken und Informationssysteme)
- Prof. Dr. Marcel Waldvogel (Verteilte Systeme)
- Juniorprofessor Dr. Steffen Rendle (Social Network Analysis)
- Juniorprofessor Dr. Tobias Schreck (Visual Analytics)
- Juniorprofessor Dr. Michael Grossniklaus (Datenbanksysteme/Database Systems)
- Dr. habil Sabine Cornelsen (Algorithmik)
- Dr. habil Sven Kosub (Formale Grundlagen)

Der Fachbereich gilt international als Zentrum der Spitzenforschung im Bereich der Datenanalyse und -visualisierung. Dazu trägt auch das im gleichen Themenumfeld angesiedelte (DFG-) Graduiertenkolleg bei, in dem einschlägige Forschung und strukturierte Doktorandenausbildung vereint sind.

3 Musterstudienpläne

Der Studienplan für das Bachelor- und Masterstudium Information Engineering zeichnet sich durch die breit gefächerten Möglichkeiten zur Vertiefung innerhalb der Kernthemen der Informatik aus. Aus allen zwölf Forschungsfeldern der Arbeitsgruppen werden Vertiefungskurse angeboten.



3.1 Bachelor Information Engineering

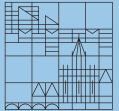
Semester	Module	Umfang in SWS	ECTS- Credits	Leistung
Grundstudium				
1	Modul Informatik 1, bestehend aus:			
	– Konzepte der Informatik	5	6	P(*)
	– Programmierkurs 1 (imperative Sprache)	4	6	S
1	Mathematische Grundlagen der Informatik	6	9	P
1	Rechnersysteme und -netze	5	6	P
2	Modul Informatik 2, bestehend aus:			
	– Konzepte der Programmierung	5	4	P
	– Programmierkurs 2 (deklarative Sprache)	4	5	S
2	Diskrete Strukturen	6	9	P
2	Datenbanksysteme	6	9	P(*)
3	Algorithmen und Datenstrukturen	6	9	P
3	Software Engineering	5	5	P
3	Statistik oder Numerik (wahlweise)	4	6	P
3	Analyse und Visualisierung von Information	6	9	P
4	Software-Projekt	4	6	P
4	Computergrafik und interaktive Systeme	6	9	P
4	Theoretische Informatik	6	9	P
1–4	Fachfremde Module und Schlüsselqualifikationen	8	13	S
Summen Pflichtveranstaltungen Grundstudium		86	120	
Vertiefungsstudium				
5	Bachelorprojekt	4	9	P
5–6	Seminar zum Bachelorprojekt	2	4	P
5–6	Vertiefende Lehrveranstaltungen	14	21	P
5–6	Fachfremde Module	7	11	S
6	Bachelorarbeit mit Kolloquium	–	15	P
Summe Vertiefungsstudium		27	60	
Gesamtsummen		113	180	

(*) Diese Veranstaltungen sind Teil der Orientierungsprüfung

P = Studienbegleitende Prüfungsleistung

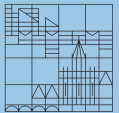
S = Studienleistung

Die Angabe von Semesterwochenstunden (SWS) ist unverbindlich. Sie dient als Hinweis auf den zu erwartenden Umfang des Präsenzstudiums.



3.2 Master Information Engineering

Semester	Lehrveranstaltungen	Umfang in SWS	ECTS- Credits
Wahlpflichtbereich			
1-3	Masterprojekt	4	9
1-3	Seminar	2	4
letztes	Masterarbeit mit Kolloquium	-	30
Summen Pflichtbereich		6	43
Wahlbereich A: Studierende mit einem mindestens vierjährigen abgeschlossenen Hochschulstudium in Information Engineering, Informatik oder einem verwandten Fach			
1	Vertiefende Lehrveranstaltungen aus dem Angebot des Fachbereichs Informatik und Informationswissenschaft und äquivalente Veranstaltungen	12	17
Summen Wahlbereich A		12	17
Wahlbereich B: Studierende mit einem weniger als vierjährigen abgeschlossenen Hochschulstudium in Information Engineering, Informatik oder einem verwandten Fach			
1-3	Vertiefende Lehrveranstaltungen aus dem Angebot des Fachbereichs Informatik und Informationswissenschaft und äquivalente Veranstaltungen	40	60
1-3	Vertiefende Lehrveranstaltungen aus dem Angebot aller Fachbereiche	12	17
Summen Wahlbereich B		52	77
Wahlbereich C: Studierende ohne Hochschulabschluss in Information Engineering, Informatik oder einem verwandten Fach			
1-3	Vertiefende Lehrveranstaltungen aus dem Angebot des Fachbereichs Informatik und Informationswissenschaft oder äquivalente Veranstaltungen	52	77
Summen Wahlbereich C		52	77
Gesamtsumme Pflichtbereich und Wahlbereich		18 bzw. 58	60 bzw. 120



3.3 Bachelor Informatik

Semester (Winter- Anfänger)	Semester (Sommer- Anfänger*)	Module	Umfang in SWS	ECTS- Credits	Leistung
Grundstudium					
1	2	Informatik 1, bestehend aus: – Konzepte der Informatik (V+Ü) – Programmierkurs 1 (imperative Sprache)	9 5 4	12 6 6	OP S
1	2	Rechnersysteme und -netze (V+Ü)	5	6	ZP
1	2	Mathematik: Analysis 1 (V+Ü)	6	9	ZP
1	2	Schlüsselqualifikationen 1	2	3	S
2	3	Informatik 2, bestehend aus: – Konzepte der Programmierung (V+Ü) – Programmierkurs 2 (deklarative Sprache)	9 5 4	9 4 5	ZP S
2	1	Datenbanksysteme (V+Ü)	6	9	OP
2	1	Mathematik: Diskrete Strukturen (V+Ü)	6	9	ZP
2	1	Schlüsselqualifikationen 2	2	3	S
3	4	Informatik 3, bestehend aus: – Betriebssysteme, Systemnahe Programmierung (V+Ü) – Programmierkurs 3 (systemnahe Sprache)	8 4 4	7 4 3	ZP S
3	4	Software Engineering (V+Ü)	5	5	ZP
3	4	Algorithmen und Datenstrukturen (V+Ü)	6	9	ZP
3	4	Mathematik: Lineare Algebra (V+Ü)	6	9	ZP
4	5	Software-Projekt	4	6	ZP
4	1	Theoretische Informatik (V+Ü)	6	9	ZP
4	3	Computergrafik und interaktive Systeme (V+Ü)	6	9	ZP
4	3	Mathematik: Statistik oder Numerik (V+Ü)	4	6	ZP
Summen Grundstudium			96	120	
Vertiefungsstudium					
5	6	Analyse und Visualisierung von Information (V+Ü)	6	9	P
5–8	5–7	vertiefende Lehrveranstaltungen (Informatik), darunter mind. 2 Seminare (à 4Cr.) und 1 Projekt (à 9 Cr.)		30	P
5–8	3, 5–8	fachfremde Lehrveranstaltungen		30	S
5–8	5–8	Schlüsselqualifikationen/Berufsvorbereitung		6	S
6/7	6/7	Praktikum oder Auslandsaufenthalt		30	N
8	8	Bachelorarbeit (12 Cr.) und Kolloquium (3Cr.)		15	P
Summe Vertiefungsstudium				120	
Gesamtsumme				240	

* = die Semesteraufteilung für Studierende, die zum Sommersemester beginnen, erfolgt individuell aufgrund von Vorkenntnissen und/oder Neigungen

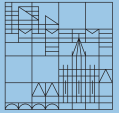
OP = Teil der Orientierungsprüfung

P = studienbegleitende Prüfungsleistung

ZP = studienbegleitende Studien- oder Prüfungsleistung und Teil der Zwischenprüfung

S = Studienleistung

N = Nachweis



4 Seminare und Projekte

4.1 Allgemeine Seminarbeschreibung

Credits

2 SWS, 4 ECTS

Semesterzuordnung

Seminare werden in jedem Semester angeboten.

Inhalte

Im Seminar wird unter Anleitung ein wissenschaftlicher Vortrag über ein gegebenes Thema vorbereitet und gehalten. Von den Teilnehmern des Seminars werden Fragen gestellt. In der Regel wird darüber hinaus eine eigenständige Ausarbeitung des Vortrags verlangt, dies kann in Form eines eigenen wissenschaftlichen Artikels geschehen.

Seminare werden von allen Arbeitsgruppen des Fachbereichs angeboten. Die Themen stammen beispielhaft aus den Bereichen:

- Algorithmik
- Bioinformatik
- Computergrafik und Visualisierung
- Datenbanksysteme
- Datenanalyse und Visualisierung
- Data Mining
- Formale Grundlagen
- Mensch-Computer-Interaktion
- Multimedia Signalverarbeitung
- Social Network Analysis
- Software Engineering
- Visual Analytics
- Verteilte Systeme

Lernziele

Der Student ist in der Lage, eigenständig eine wissenschaftliche Präsentation auszuarbeiten, vorzutragen und Fragen zu beantworten.

Voraussetzung

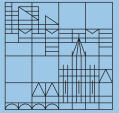
In der Regel keine. Ein Seminar kann begleitend zu einem Bachelor- oder Masterprojekt angeboten werden.

Arbeitsaufwand

120 Stunden, davon ca. 30 Stunden Präsenzstudium und ca. 90 Stunden Selbststudium

Literatur

Wird im jeweiligen Seminar individuell ausgegeben



4.2 Allgemeine Bachelor-Projektbeschreibung

Credits

2 SWS, 9 ECTS

Semesterzuordnung

BA-Projekte werden in jedem Semester angeboten.

Inhalte

Im Projekt wird auf das eigenständige wissenschaftliche Arbeiten vorbereitet. Die Teilnehmer erhalten Anleitung im wissenschaftlichen Schreiben, üben Literaturarbeit und arbeiten sich in das Themengebiet der Bachelorarbeit ein.

Lernziele

Absolventen sind in der Lage, grundlegende wissenschaftliche Arbeit zu verrichten.

Arbeitsaufwand

300 Stunden

Literatur

Wird im jeweiligen Projekt individuell ausgegeben

4.3 Allgemeine Master-Projektbeschreibung

Credits

2 SWS, 9 ECTS

Semesterzuordnung

MA-Projekte werden in jedem Semester angeboten.

Inhalte

Das Projekt bereitet auf die Masterarbeit vor. Die Teilnehmer erhalten Anleitung im wissenschaftlichen Schreiben und in der Präsentation eigener wissenschaftlicher Beiträge, üben den vertieften Umgang mit Literaturarbeit und arbeiten sich in das Themengebiet der Masterarbeit ein.

Lernziele

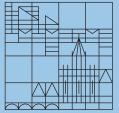
Absolventen sind in der Lage, eigenständig kleinere wissenschaftliche Arbeiten zu verrichten.

Arbeitsaufwand

300 Stunden

Literatur

Wird im jeweiligen Projekt individuell ausgegeben



5 Information Engineering and Informatics

5.1 Definition

Information Engineering at the University of Konstanz encompasses all aspects of the mining, preparation, selection, extraction, analysis and communication of information. We understand it as applied computer science, which addresses all the tasks associated with the data-information-knowledge process chain.

Information Engineering focuses on the conceptual design and construction of information-processing systems and the respective methods and applications. Examples of relevant topics are efficient coding methods for transferring multimedia data, query evaluation in XML databases, data mining capabilities in drug production or the visualisation of weather data. Further relevant topics are the user-friendliness of computer programmes and Web design as well as ethical issues relevant to the availability of information. The course therefore combines fundamental issues of practical and applied informatics with the user-oriented approaches of information science.

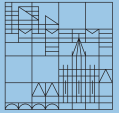
The study programme is based on three pillars (in this order):

1. Informatics
2. Information Handling
3. Mathematics

It focuses particularly on the area of data exploration and visualisation. Our teaching is closely in touch with the research conducted in the department and true to the principle of the University of Konstanz: „**Teaching from research**“. The department’s research focus, „**Visualisation and Exploration of Large Information Spaces**“ gives its research a unique breadth in Germany and enables the department to enjoy international standing.

The course is complemented by additional study in a field not directly related to the student’s degree course (non-specialist). The intention here is to prepare students for the fact that their postgraduate work will include handling information requirements from other areas of application. The need to manage increasing volumes of information emerging from business, public services and society is opening a wide range of opportunities for graduates to put their acquired skills into practice. In addition to the many jobs directly related to computer science, typical professions include work in the area of information services.

The objective behind classic **informatics** is to provide a degree course that is oriented towards the basic principles and scopes of application, involving the entire spectrum of information processing systems and machines. In contrast to other universities and colleges, our degree course also provides knowledge about theoretical concepts and methods which are separate from current trends. In their professional lives, computer scientists will find that this will give them considerable independence from industry trends. The proportion of maths and theory is therefore greater in this course. As this is an eight-term degree programme, the curriculum provides sufficient time for a practical phase (one term spent abroad or a term spent working in industry with experience gained abroad preferred).



5.2 Structure of the degree course

Information Engineering has been offered at the University of Konstanz as the first consecutive Bachelor and Master degree programme since the winter term 1999/2000. In the course of the accreditation process, conducted by ASIN, a professional agency, in 2005 (this was another first at the University of Konstanz), the study programme was reviewed in the winter term 2006/2007 and again in the winter term 2010/2011 and improved. The following information describes the degree courses. **Binding information is documented in the respectively applicable examination regulations for Bachelor and Master degree courses.**

Both Bachelor degree courses are generally held in German, however knowledge of English will be required for in-depth coursework programmes; course-related literature is often in English.

The Information Engineering Bachelor degree course comprises three years, while a Bachelor degree in Informatics comprises four years. Both impart the basic knowledge and skills to enable graduates to take up typical professions in today's information society.

Straight after the Bachelor degree course, or after a number of years of professional experience, or even after a completely different degree course, in-depth knowledge can be acquired within the framework of the Master degree course. How long the Master degree course takes depends on the Bachelor degree. In combination with three-year Bachelor degree courses, the Master degree takes two years; following a four-year, equivalent Bachelor degree course, the Master degree only takes one year.

Each year of study is divided into two halves: the winter and summer terms. Apart from the final examination, all other examinations are related to coursework, meaning that the content of the course is tested at the end of the term in which the course module has been taken. Examinations are in the form of either assignments, oral examinations or a written examination.

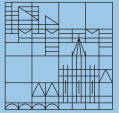
The work involved in each course is measured in credits in accordance with the ECTS (European Credit Transfer System). According to this system, one credit represents 30 hours of work. Both degree courses require that you collect 30 ECTS credits per term.

5.3 Qualification targets of the Information Engineering Bachelor Degree Course

The *Information Engineering* degree course provides students with the skills to be just that - an information engineer who, working from a use-oriented perspective in the large information spaces that characterise our information society, searches, filters and organises data and prepares, summarises and presents it as information. Furthermore, the course teaches the methods, applications and evaluations of systems in automatic information processing based on the principles of computer science and mathematics.

The Information Engineering Bachelor degree course teaches the principles required to collect and prepare data from a use-oriented perspective in order to extract information and present it in such a way as to be able to competently assess and apply the knowledge that has been thus gained. The combination here of theoretical and practical informatics with use-oriented methods of information science is the objective of the course.

When the degree course was developed, particular attention was paid to teaching theoretical, basic knowledge on the one hand but also on ensuring that the degree course includes sufficient practical orientation to qualify students for a job.



Skills. When the degree course was developed, particular attention was paid to ensuring that skills are taught that will be required when using information systems in modern organisations, taking technical, administrative and strategic aspects into consideration. Graduates should be able to work in information technology institutions that are responsible for providing information products.

Graduates will have fundamental, scientific and technical knowledge and will have an overview of the field of Information Engineering.

5.4 Structure of the Information Engineering Bachelor degree course

The Information Engineering Bachelor degree course combines the principles of information processing with in-depth courses offered by the department in a range of appealing fields, but particularly in the field of exploration and visualisation of information.

Admission to the degree programme is regulated by the Admissions and Matriculations policies of the University of Konstanz. As there are no further restrictions over and above the specified conditions, the general higher education entrance qualification are all that is required.

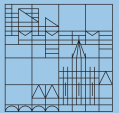
The Bachelor degree course comprises a four-term Basic Study Period followed by a two-term Advanced Coursework Programme. The following required courses have to be taken during the Basic Study Period:

Informatics

- Informatics Module I (concepts of computer science, programming course 1)*
- Informatics Module 2 (concepts of programming and programming course 2)
- Computer Systems and Networks
- Algorithms and Data Structures
- Theoretical Informatics
- Software Engineering
- Software Project

Information Handling

- Database Systems*
- Analysis and Visualisation of Information
- Computer Graphics and Interactive Systems



Mathematics

- Mathematic Principles in Computer Science
- Discrete Structures
- Statistics or Numerics

Those courses marked with an * have to be taken and passed by the end of the second term in order to pass the **orientation test**.

With regard to the **non-specialist modules**, there is a list of suggested courses, which highlights sensible combinations that have proven their worth. It is also possible to take courses for key qualifications offered by a central body.

The **Advanced Coursework Programme** enables students to choose courses from a variety of areas of research relevant to Information Engineering and is very project oriented. The Bachelor project is the focus of the Advanced Coursework Programme. Over a period of four months in their final term, students are required to prepare a scientific thesis - their Bachelor thesis - the content of which is examined and questioned in the form of an oral examination in a colloquium.

On a three-year Bachelor degree course, 120 ECTS credits have to be gained during the Basic Study Period, followed by 60 ECTS credits during the Advanced Coursework Programme.

The final grade is made up (in varying proportions) from the grades acquired from coursework, from the final Bachelor thesis and from the oral colloquium examination.

Successful students are awarded the academic title of **Bachelor of Science (B.Sc) in the field of Information Engineering**. They are fit to tackle the diverse assignments in today's information society and have the skills needed to start their professional lives. If they wish, they can expand their knowledge and expertise in the context of a Master degree course in Information Engineering or in another related Master degree course.

5.5 Qualification targets of the Informatics Bachelor degree course

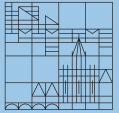
The aim of the Informatics Bachelor degree course is to teach the principle methods of applied computer science and impart a solid foundation in theoretical and mathematical skills.

Informatics plays a key role in a steadily increasing number of industries. The high level of relevance of this area can be seen by the fact that leading companies in all industries have identified informatics as a strategic focus of their activities. The Informatics Bachelor degree course portfolio concentrates on the expertise that is required by potential employers.

The biggest target group for this course is school-leavers who are interested in a Bachelor study course in Informatics and would like to benefit from the research focus of the Department of Informatics and Information Science and the external practical course-component, spent either abroad or in industry. → The degree enables direct employment in a promising segment of the employment market or admission to a higher degree or PhD course.

These new students distinguish themselves from students on the Information Engineering Bachelor degree course by the fact that they will have a greater interest in mathematical and system-related principles.

The degree course



1. qualifies students for jobs (rather than merely enabling them)
2. comprises four years including a one-term external practical component,
3. offers a fast-track option with subsequent admission to a PhD course of study,
4. includes a mentor programme, which intends to help make studying easier for students, and
5. has a high degree of permeability to and from the existing Bachelor degree course in Information Engineering and the module-based teacher training, where Informatics is the core subject.

Skills. Graduates in Informatics are able to deal with growing volumes of data in processing systems and in addition to the acknowledged ability to produce and audit new software, they have the methodological competence required to explore and visualise information spaces as a basis for making decisions.

The academic training profile of graduates is therefore not only an academic research profile but one that responds to the demands placed by industry. This kind of profile is also required in the research and development labs of the manufacturing industry and information markets.

5.6 Structure of the Informatics Bachelor degree course

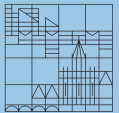
The Informatics Bachelor degree course focuses on a wider basis in the areas of mathematics and informatics principles. In addition to this, the eight terms enable students to include a practical component or term spent abroad, which earns the student 30 ECTS in accordance with the examination regulations. In the Advanced Coursework Programme, an additional 30 ECTS are gained by completing work in the research areas of the Department of Informatics and Information Science.

Admission to the degree programme is regulated by the Admissions and Matriculations policies of the University of Konstanz. As there are no further restrictions over and above the specified conditions, the general higher education entrance qualification is all that is required.

The Bachelor degree course comprises a four-term Basic Study Period followed by a four-term Advanced Coursework Programme. The following required courses have to be taken during the Basic Study Period:

Informatik

- Informatics Module I (concepts of computer science, programming course 1)*
- Informatics Module 2 (concepts of programming and programming course 2)
- Informatics Module 3 (operating systems and system-relevant programming and programming course 3)
- Computer Systems and Networks
- Algorithms and Data Structures



- Theoretical Informatics
- Software Engineering
- Software Project

Information Handling

- Database Systems*
- Analysis and Visualisation of Information
- Computer Graphics and Interactive Systems

Mathematics

- Analysis 1
- Linear Algebra 1
- Discrete Structures
- Statistics or Numerics

Those courses marked with an * have to be taken and passed by the end of the second term in order to pass the **orientation test**. Within the four-year Bachelor course, an **Intermediate Examination** (part of coursework; no additional examination) has to be passed by students to the extent that they complete the fourth term of study.

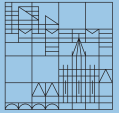
Non-specialist **modules** are taken during the Advanced Coursework Programme; key qualifications are gained during the Basic Study Period and the Advanced Coursework Programme.

The **Advanced Coursework Programme** brings both Bachelor degree courses closer together. Informatics students attend courses from the different areas of research of the field of Information Engineering as part of their Elective Study modules. The Bachelor project is completed between terms 5 and 7. During the 8th term, students complete their Bachelor thesis, the content of which is examined and questioned in the form of an oral examination in a colloquium.

In the Basic Study Period and in the Advanced Coursework Programme, students are required to achieve 120 ECTS credits or a total of 240 ECTS credits at the end of both components.

The final grade is made up (in varying proportions) from the grades acquired from coursework, from the final Bachelor thesis and the oral colloquium examination.

Successful students are awarded the academic title of **Bachelor of Science (B.Sc) in the field of Informatics (Computer Science)**. Graduates are prepared for a wide range of tasks in today's global information society. The four-year degree qualifies for jobs and gives students initial work experience thanks to the practical component of the course. For admission to a Master degree course, this four-year degree entitles graduates to be admitted to the **shorter Master degree course**, which comprises one year. It is also possible to be **admitted directly to a PhD course**.



5.7 Qualification targets of the Information Engineering Master degree course

The *Information Engineering* degree course provides students with the skills to be just that - an information engineer who, working from a use-oriented perspective in the large information spaces that characterise our information society, searches, filters and organises data and prepares, summarises and presents it as information. Furthermore, the course communicates the methods, applications and evaluations of the systems in automatic information processing based on the principles of computer science and mathematics.

The Master degree course is aimed at international graduates who have completed Bachelor degrees in Information Engineering and Informatics (Computer Science) as well as suitably qualified people from other fields, who have an above-average university degree in a related subject.

The objective of this degree course is to take the fundamental knowledge gained from the Bachelor degree in Information Engineering or Informatics (Computer Science) and to go into more diversified and advanced levels.

The respective study modules are geared towards enabling students to apply scientific methods and insights to difficult and complex computer-science problems both in practice and in research. In terms of its direction of focus, this Master degree course belongs to the more research-oriented degree programmes. Teaching is conducted by lecturers and associate professors and is particularly based on their experiences from current research. Students are integrated at an early stage into on-going research projects as part of the required internship and expand on this knowledge later during the Master thesis.

Skills. Through the Master degree, graduates gain in-depth scientific knowledge about information engineering and are able to work independently according to scientific principles and apply scientific methods and insight.

The Master degree courses are held in English. The aim here is to prepare graduates for careers with internationally-operating companies, projects and further postgraduate study.

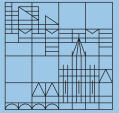
Graduates are specialists in the use-oriented processing and preparation of information.

5.8 The structure of the Master's degree course in Information Engineering

The Master's degree course in Information Engineering expands on the knowledge gained from the Bachelor degree programme and goes into computer science in terms of the exploration of data, visualisation and the respective forms of human interaction.

Admission to the Master's degree programme is subject to the admission regulations. Successful applicants have normally completed related study programmes (Bachelor degrees in Information Engineering and Informatics) with a good grade. Graduates from other fields may also be admitted if they can prove that they have other eligible qualifications or the required prior knowledge by taking an approval test.

The Master degree course comprises a compulsory elective study component and an elective study component, during which the final Master thesis is prepared and subsequently presented in a colloquium (oral examination). In this **compulsory elective component**, students are required to choose an Information Engineering research area as their **area of specialisation**. Based on



this area of specialisation, additional in-depth courses from the elective study part are chosen, enabling students to put their coursework together very individually. Students from other fields of study expand their knowledge more broadly by choosing courses from the extensive elective study component of Information Engineering and do not have the option of being able to take courses in other subjects (non-specialist subjects) to the extent of 17 ECTS credits.

How long the elective study period lasts depends on whether students have been admitted to a one or two-year Master degree course. In the event of a one-year Master degree course, the course begins with the Master project with in-depth courses (amounting to 17 ECTS credits) being taken at the same time. The Master thesis is completed during the second term and is concluded by the oral colloquium examination. In the event of a two-year Master degree course, students can spend three terms on their compulsory electives and elective study components. Students are required to gain 60 or 77 ECTS credits by taking in-depth courses. Non-specialist courses are optionally available to groups (with an equivalent Bachelor degree), whereby 17 ECTS credits can be acquired.

The key to how the study programme is planned is a compulsory Mentor meeting, which takes place during the first term. The subject of this meeting is the student's individual study plan and the definition of potential study profiles. A study profile can include an additional specialisation. Study profiles are optional and are accredited in the final certificate.

The final grade is made up (to varying proportions) of grades from coursework, the Master thesis and the oral colloquium examination.

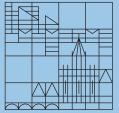
Successful graduates are awarded the title of **Master of Science (M. Sc.) in Information Engineering**. They are experts in processing information and qualified to handle advanced tasks in today's information society. They can expand their scientific specialisation in the context of a PhD programme in Informatics or a related subject. A possible study profile can be added to the degree certificate in the form of a specialisation.

Study profiles are **models** which help Master students to shape their own individual study focus. They give the Master degree course in Information Engineering a specialisation in the wide field of applied and practical computer science:

- Visual Computing
- Human-Computer Interaction
- Digital Libraries
- Data Mining
- Information Systems Engineering

6 Department of Informatics and Information Science

With its nine professors, three junior professors, lecturers and 80 research assistants, the department amply covers the spectrum of knowledge required for a degree in informatics, and is uniquely and excellently staffed with respect to its research focus of the **exploration and visualisation of large volumes of data**.

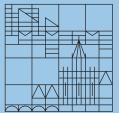


- Prof. Dr. Michael Berthold (Bioinformatics & Information Mining)
- Prof. Dr. Ulrik Brandes (Algorithmics)
- Prof. Dr. Oliver Deussen (Computer Graphics & Media Design)
- Prof. Dr. Daniel Keim (Databases, Data Mining & Visualisation)
- Prof. Dr. Stefan Leue (Software Engineering)
- Prof. Dr. Harald Reiterer (Human-Computer Interaction)
- Prof. Dr. Dietmar Saupe (Multimedia Signal Processing)
- Prof. Dr. Marc Scholl (Databases and Information Systems)
- Prof. Dr. Marcel Waldvogel (Distributed Systems)
- Juniorprofessor Dr. Steffen Rendle (Social Network Analysis)
- Juniorprofessor Dr. Tobias Schreck (Visual Analytics)
- Juniorprofessor Dr. Michael Grossniklaus (Database Systems)
- Dr. habil Sabine Cornelsen (Algorithmics)
- Dr. habil Sven Kosub (Formal Foundations)

The department has an international reputation as a centre for cutting-edge research in the field of data analysis and visualisation. The German Research Foundation (DFG)-based PhD Graduate School, where unique research and structured doctoral level training are united, also contributes to the excellent repute of the department.

7 Sample study plans

The study plan for Bachelor and Master degree courses in Information Engineering is distinguished by the wide range of opportunities it offers to specialise in the core topics of informatics. In-depth courses are offered in all twelve areas of research conducted by the work groups.



7.1 Bachelor in Information Engineering

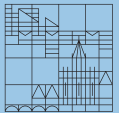
Semester	Modules	Extent in SWS	ECTS credits	Examination
Basic Study Period				
1	Informatics Module 1, consisting of:			
	– The Concepts of Informatics	5	6	P(*)
	– Programming Course 1 (imperative Programming Language)	4	6	S
1	Mathematic Principles in Computer Science	6	9	P
1	Computer Systems and Networks	5	6	P
2	Informatics Module 2, consisting of:			
	– The Concepts of Programming	5	4	P
	– Programming Course 2 (imperative Programming Language)	4	5	S
2	Discrete Structures	6	9	P
2	Database Systems	6	9	P(*)
3	Algorithms and Data Structures	6	9	P
3	Software Engineering	5	5	P
3	Statistics or Numerics (option)	4	6	P
3	Analysis and Visualisation of Information	6	9	P
4	Software Project	4	6	P
4	Computer Graphics und Interactive Systems	6	9	P
4	Theoretical Informatics	6	9	P
1–4	Non-specialist Modules plus Key Qualifications	8	13	S
Total Required Courses during Basic Study Period		86	120	
Advanced Coursework Program				
5	Bachelor Project	4	9	P
5–6	Seminar related to Bachelor Project	2	4	P
5–6	Advanced Specialist Lectures	14	21	P
5–6	Non-specialist Modules	7	11	S
6	Bachelor Thesis plus Colloquium	–	15	P
Total Advanced Coursework		27	60	
Overall Total		113	180	

(*) These courses are part of the preliminary examination

P = course-related examination (credit points)

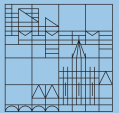
S = study credits

The specification of semester week hours (SWS) is not binding. It merely serves as a means of indicating the extent of time spent on the course that is expected.



7.2 Master in Information Engineering

Semester	Lectures	Extent in SWS	ECTS credits
Compulsory Elective Study Area			
1–3	Master project	4	9
1–3	Seminar	2	4
letztes	Master Thesis plus Colloquium	–	30
Total Compulsory		6	43
Elective Area A: Students who have completed at least a four-year university degree in Information Engineering or a related subject			
1	Advanced lectures and courses on topics offered by the Department of Informatics and Information Science as well as associated courses.	12	17
Total Elective Study Area A		12	17
Elective Study Area B: Students who have completed a university degree in Information Engineering, Informatics or a related subject which comprised less than four years			
1–3	Advanced lectures and courses on topics offered by the Department of Informatics and Information Science as well as associated courses	40	60
1–3	Advanced or non-specialist lectures on topics offered by all departments	12	17
Total Elective Study Area B		52	77
Wahlbereich C: Elective Study Area C: Students who have not completed a university degree in Information Engineering, Informatics or a related subject			
1–3	Advanced lectures and courses on topics offered by the Department of Informatics and Information Science as well as associated courses.	52	77
Total Elective Study Area C		52	77
Overall Total of Compulsory Study Area + Compulsory Elective Study Area		18 bzw. 58	60 bzw. 120



7.3 Bachelor in Informatics

Winter term (entry)	Summer term (entry)	Modules	Scope in SWS	ECTS Credits	Examination
Basic Study Period					
1	2	Informatics 1, comprising of: – Concepts of Informatics (*) – Programming Course 1 (imperative programming language)	9 5 4	12 6 6	OP S
1	2	Computer Systems and networks (lecture+tutorial)	5	6	ZP
1	2	Analysis 1 (L+T)	6	9	ZP
1	2	Key Qualifications 1	2	3	S
2	3	Informatics 2, comprising of: – Concepts of Programming (L+T) – Programming Course 2 (declarative programming language)	9 5 4	9 4 5	ZP S
2	1	Database Systems (L+T)	6	9	OP
2	1	Discrete Structures (L+T)	6	9	ZP
2	1	Key Qualifications 2	2	3	S
3	4	Informatics 3, comprising of: – Operating Systems, System-relevant Programming (L+T) – Programming Course 3 (system-relevant programming language)	8 4 4	7 4 3	ZP S
3	4	Software Engineering (L+T)	5	5	ZP
3	4	Algorithms and Data Structures (L+T)	6	9	ZP
3	4	Linear Algebra (L+T)	6	9	ZP
4	5	Software Project	4	6	ZP
4	1	Theoretical Informatics (L+T)	6	9	ZP
4	3	Computer Graphics and Interactive Systems (L+T)	6	9	ZP
4	3	Statistics or Numerics (L+T)	4	6	ZP
Total Basic Study Period			96	120	
Advanced Coursework Programme					
5	6	Analysis and Visualisation of Information (L+T)	6	9	P
5–8	5–7	advanced Specialist Lectures (Informatics), of these, at least 2 seminars (à 4Cr.) and 1 Project (à 9 Cr.)		30	P
5–8	3, 5–8	Lectures held by other departments (non-specialist)		30	S
5–8	5–8	Key Qualifications/Job Preparation		6	S
6/7	6/7	Internship or Study Abroad		30	N
8	8	Bachelor Thesis (12 Cr.) and Colloquium (3Cr.)		15	P
Total Advanced Coursework				120	
Overall Total				240	

* = Term organisation for students who start their course in the summer term is handled individually according to prior knowledge and/or interests

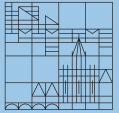
OP = Part of Orientation Test

P = Course-related examination

ZP = Course-related study credits or examination and Part of intermediate Examination

S = study credits

N = Evidence



8 Seminars and Projects

8.1 Seminars

Credits

2 SWS, 4 ECTS

Assignment to semesters

Seminars are offered every semester.

Topics

Within a seminar, students prepare a scientific talk about a given topic. Supervision is granted. After the talk other participants of the seminary may ask questions. Moreover, a written report about the topic has to be prepared. This can be done in form of a scientific article.

Seminars are offered every semester from all work groups of the department. The subjects are e.g.:

- Algorithmics
- Human-Computer Interaction
- Bioinformatics & Information Mining
- Multimedia Signal Processing
- Computer Graphics & Media Design
- Software Engineering
- Database and Information Systems
- Distributed Systems
- Data Analysis and Visualisation
- Visual Computing
- Formal Foundations
- Social Network Analysis
- Visual Analytics

Learning objectives

Participants are able to prepare a scientific presentation, to present it in a talk and to answer questions.

Requirements

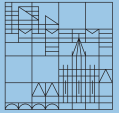
Generally none. A seminar can be offered in combination with a bachelor or master project.

Amount of work

120 hours, ca. 30 hours learning in classes, ca. 90 hours home work

Literature

Will be distributed individually for each seminar



8.2 Bachelor Projects

Credits

2 SWS, 9 ECTS

Assignment to semesters

Bachelor projects are offered every semester.

Topics

Within the project participants will be prepared for individual scientific work. They receive advice for scientific writing and work with scientific literature. Furthermore they familiarize with the thematic topic of the planned bachelor thesis.

Learning objectives

Participants are able to perform basic scientific work.

Requirements

Lectures from compulsory study area

Amount of work

300 hours

Literature

Will be distributed individually for each project.

8.3 Master Projects

Credits

2 SWS, 9 ECTS

Assignment to semesters

Master projects are offered every semester

Topics

The project prepares for the master's thesis. Participants will receive advice for individual scientific work and for the presentation of their own scientific results. They also receive advice for improved scientific writing and for elaborate working with scientific literature. Furthermore they familiarize with the thematic topic of the planned masters thesis.

Learning objectives

Participants are able to individually perform smaller scientific works.

Requirements

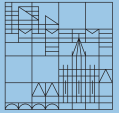
Lectures from compulsory study area

Amount of work

300 hours

Literature

Will be distributed individually for each project



9 Module Directory / Modulverzeichnis

Following section includes detailed descriptions of the lectures. The lectures belonging to the required course component are only offered once a year. Advanced courses are subject to change. The courses listed here belong to those courses that are offered regularly and will be held in this form or very similar. They are closely related to the respective area of research of the lecturer. The list of modules is generated from the electronic course catalogue, LSF, to make sure that the list is always up to date. This way, updates can be seen online straightaway. If required, students can access a PDF, which is generated based on the currently published descriptions of the modules. This "best-practice approach" was developed as a pilot project by the Department of Informatics and Information Science.

The Student Affairs committee of the department is the body of people responsible for ensuring the quality of the offered courses. They ensure that a sufficient choice of lectures from all areas of research relating to Information Engineering is offered in each term. The current choice of lectures and courses is published in due time before the start of each term in the electronic course catalogue, LSF.

Im nächsten Abschnitt sind detaillierte Beschreibungen der Lehrveranstaltungen aufgeführt. Die Veranstaltungen des Pflichtbereichs werden grundsätzlich einmal im Studienjahr angeboten. Bei den Vertiefungsveranstaltungen ist das Angebot veränderlich, die hier aufgeführten zählen jedoch zu den immer wieder angebotenen und können in dieser oder ähnlicher Form erwartet werden. Sie sind eng an den jeweiligen Forschungsschwerpunkt der Dozenten gekoppelt. Das Modulverzeichnis wird für die Möglichkeit der laufenden Aktualisierung, aus dem elektronischen Vorlesungsverzeichnis, LSF, generiert. Auf diese Weise können Aktualisierung sofort online nachgelesen werden. Bei Bedarf kann dann ein pdf-Dokument abgerufen werden, das auf Basis der aktuell publizierten Modulbeschreibungen generiert wird. Dieser „best-practice-Ansatz“ wurde als Pilotprojekt am Fachbereich Informatik und Informationswissenschaft entwickelt.

Die Studienkommission des Fachbereichs ist das verantwortliche Gremium für die Qualitätssicherung des Lehrangebots. Sie stellt sicher, dass in jedem Semester eine ausreichende Auswahl an Veranstaltungen aus allen Forschungsgebieten des Information Engineering zur Verfügung steht. Rechtzeitig vor Beginn eines Semesters wird das jeweils aktuelle Veranstaltungsangebot im elektronischen Vorlesungsverzeichnis LSF veröffentlicht.

10 Studiumsvorbereitung

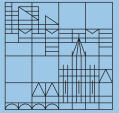
10.1 Brückenkurs Mathematik

Modulverantwortlicher : PD Dr. Kosub (z.B. INF-11690-20122)

Credits : keine

Inhalte :

Dieser einwöchige Kurs soll Studienanfänger dabei unterstützen, sich auf die Arbeitswei-



se in den Bachelor-Studiengängen Information Engineering und Informatik einzustellen. Da erfahrungsgemäß viele Studienanfänger mit mathematischen Inhalten Schwierigkeiten haben, werden Themen aus der Schulmathematik aufgegriffen und wiederholt, wobei hinsichtlich der Vorlesungen im ersten Semester Themen aus der diskreten Mathematik im Vordergrund stehen werden. Der Kurs gliedert sich in zwei Teile: einen (größeren) Vorlesungsteil und einen Übungsteil, in dem auf die vorher in der Vorlesung angesprochene Themen mit Aufgaben und Beispielen näher eingegangen wird. Dabei wird auch das eigenständige Lösen von Aufgaben Bestandteil der Übungen sein. Schwerpunkte:

- Arithmetik
- Polynome
- Folgen
- Reihen
- Kombinatorik
- Induktion
- Lineare Gleichungssysteme

Lernziele :

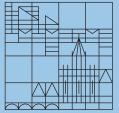
Auffrischung und Angleichung mathematischer Kenntnisse und Arbeitsweisen aus der Schule am Beispiel von Fragestellungen aus der Algorithmik sowie der diskreten Mathematik.

Voraussetzung :

Keine.

Literatur :

Es wird keine Literatur vorausgesetzt.



11 Grundstudium - 1. Semester - Pflichtbereich

11.1 Mathematische Grundlagen der Informatik (Mathematical Foundations of Computer Science)

Modulverantwortlicher : PD Dr. Kosub (z.B. INF-11920-20112)

Credits : 6 SWS, 9 ECTS

Inhalte :

- Logik (Aussagen, Quantoren, Beweise)
- Mengen (Darstellungen, Operationen, Mengenfamilien)
- Relationen (Funktionen, Äquivalenzrelationen, Ordnungsrelationen, Graphen)
- Induktion (vollständige Induktion, strukturelle Induktion, transitive Hülle, Mächtigkeit von Mengen)
- Analysis (Konvergenz von Folgen und Reihen, oberer und unterer Grenzwert, Potenzreihen, Asymptotik)
- Lineare Algebra (Vektorräume, lineare Abbildungen, Eigenwerte und Eigenvektoren)

Lernziele :

Beherrschung grundlegender Konzepte der Diskreten Mathematik und Logik, wie sie für das Information Engineering vorausgesetzt werden. Die Studierenden lernen vor allem den Umgang mit eindeutiger Notation sowie die Formulierung und den Beweis von Aussagen über formale Sachverhalte. Außerdem die Beherrschung grundlegender Techniken der Analysis und der Linearen Algebra.

Voraussetzung :

Es gibt keine Voraussetzungen im engeren Sinne, empfohlen wird jedoch die Teilnahme am Brückenkurs Mathematik.

Arbeitsaufwand :

Präsenzstudium: 84 Stunden, Eigenstudium: 186 Stunden

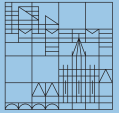
Prüfungsleistung :

Klausur, 120 Minuten, sowie 50% der Punkte aus den wöchentlichen Übungsaufgaben.

Literatur :

Es gibt keine spezifische Literaturvorgabe, empfohlen werden können aber zum Beispiel:

- Christoph Meinel, Martin Mundhenk. *Mathematische Grundlagen der Informatik. Mathematisches Denken und Beweisen. Eine Einführung. 3., überarbeitete und erweiterte Auflage.* B. G. Teubner Verlag, Wiesbaden, 2006.
- Bernd Kreuzler, Gerhard Pfister. *Mathematik für Informatiker.* Springer-Verlag, Berlin, 2009



11.2 Konzepte der Informatik (Principles of Computer Science 1)

Modulverantwortlicher : Prof. Dr. Berthold (z.B. INF-11700-20122)

Credits : 5 SWS, 6 ECTS

Inhalte :

Das Modul Informatik 1 besteht aus der Vorlesung “Konzepte der Informatik“ und dem “Programmierkurs 2“. Es empfiehlt sich deswegen auch für Nebenfächler, beide Veranstaltungen zu belegen. Kern der Vorlesung “Konzepte der Informatik“ sind die folgenden Themen:

- Objektorientierte Programmierung - mit Java als Programmiersprache werden Konzepte objektorientierter Programmiersprachen wie Klassen, Vererbung, Polymorphismus, Ausnahmebehandlung oder generische Programmierung vermittelt (ca. 10 + 6 SWS)
- Algorithmen und Datenstrukturen - häufig verwendete Datenstrukturen wie Listen, Bäume, Streuspeicherung oder Graphen sowie rekursive Algorithmen und Sortierverfahren wie Quicksort, Mergesort oder Heapsort werden behandelt (ca. 18 + 8 SWS)
- Theoretische Grundlagen - Fragen der Berechenbarkeit von Problemen, Komplexität und Korrektheit von Algorithmen; Einführung in die Automatentheorie sowie formale Sprachen und Grammatiken (ca. 12 + 6 SWS)
- Angewandte Programmierung - Analyse von nebenläufigen Prozessen, Entwicklung von einfachen parallelen Algorithmen und Diskussion von Synchronisationsproblemen; Implementierung von graphischen Benutzeroberflächen (ca. 8 + 4 SWS)

Lernziele :

Absolventen haben ein grundlegendes Verständnis der objektorientierten Programmierung und Modellierung. Grundlegende Datenstrukturen (u.a. Listen, Bäume, Hashes) und Algorithmen (u.a. Sortieren) sind ebenso verstanden wie wichtige theoretische Konzepte (u.a. Berechenbarkeit, Komplexität, formale Sprachen und Grammatiken).

Voraussetzung :

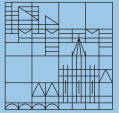
Keine

Arbeitsaufwand :

Präsenzstudium: 70 Stunden, Eigenstudium: 110 Stunden

Prüfungsleistung :

- Studienleistung: 60% der Punkte aus den Übungen, mindestens 40% pro Aufgabenblatt
- Prüfungsleistung: Klausur von 120 Minuten Dauer, Teilnahmevoraussetzung ist das Absolvieren der Studienleistung
- Die Note entspricht der Klausurnote



Literatur :

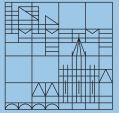
Leider gibt es kein einzelnes Buch, das alle Inhalte der Vorlesung behandelt. Deswegen ist die folgende Liste eine Sammlung an Büchern, die die meisten relevanten Themen enthalten.

Grundlagen der Informatik:

- Herold, Lurz & Wohlrab: *Grundlagen der Informatik* (lbs 830/h27, ISBN 3-8273-7305-2)
- Grumm & Sommer: *Einführung in die Informatik* (lbs 830/g95(7), ISBN 3-486-58115-7)
- Küchlin & Weber: *Einführung in die Informatik* (lbs 843/k92(3), ISBN 3-540-20958-1)

Algorithmen und Datenstrukturen:

- Cormen: *Algorithmen - eine Einführung* (lbs 840/a53, ISBN 3-486-58262-8) oder das englische Original *Introduction to Algorithms* (lbs 830/c67(28), ISBN 0-262-03293-7)
- Sedgewick: *Algorithmen in Java* (Teil 1-4) (kid 112:n/s26-1/4, ISBN 3-8273-7072-8) oder das englische Original inklusive Graphalgorithmen: *Algorithms in Java, Part 1-4* (ISBN 0-201-36120-9), *Algorithms in Java, Part 5* (ISBN 0-201-36121-6)
- Ottmann & Widmayer: *Algorithmen und Datenstrukturen* (lbs 830/o99(4), ISBN 3-8274-1029-0)



11.3 Programmierkurs 1 (Programming Course 1)

Modulverantwortlicher : Prof. Dr. Berthold (z.B. INF-11930-20122)

Credits : 4 SWS, 6 ECTS

Inhalte :

Das Modul Informatik 1 besteht aus der Vorlesung “Konzepte der Informatik“ und dem “Programmierkurs 2“. Es empfiehlt sich deswegen auch für Nebenfächler, beide Veranstaltungen zu belegen. Kern des “Programmierkurs 1“ sind die folgenden Themen:

- Objektorientierte Programmierung - die in der Vorlesung “Konzepte der Informatik“ vorgestellten Konzepte objektorientierter Programmiersprachen wie Klassen, Vererbung, Polymorphismus, Ausnahmebehandlung oder generische Programmierung werden praktisch mit Java an Hand verschiedener Beispiele geübt (ca. 6 + 6 SWS)
- Algorithmen und Datenstrukturen - die in der Vorlesung vorgestellten häufig verwendeten Datenstrukturen wie Listen, Bäume, Streuspeicherung oder Graphen sowie rekursive Algorithmen und Sortierverfahren wie Quicksort, Mergesort oder Heapsort werden an Hand von Beispielen mit Java geübt (ca. 8 + 8 SWS)
- Angewandte Programmierung - Implementierung von parallelen und nebenläufigen Programmen mit Java; Verwendung von integrierten Entwicklungsumgebungen wie BlueJ oder Eclipse; Dokumentation und Testen von Programmen (ca. 8 + 8 SWS)

Lernziele :

Absolventen haben ein grundlegendes Verständnis der objektorientierten Programmierung mit Java. Grundlegende Datenstrukturen (u.a. Listen, Bäume, Hashes) und Algorithmen (u.a. Sortieren) können selbstständig implementiert werden.

Voraussetzung :

Keine

Arbeitsaufwand :

Präsenzstudium: 56 Stunden, Eigenstudium: 124 Stunden

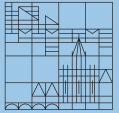
Prüfungsleistung :

- nur unbenotete Studienleistung möglich
- 60% der Punkte aus den Übungen, mindestens 40% pro Aufgabenblatt und
- 50% der Punkte aus den vier Programmiertests

Literatur :

Leider gibt es kein einzelnes Buch, das alle Inhalte der Vorlesung behandelt. Deswegen ist die folgende Liste eine Sammlung an Büchern, die die relevanten Themen enthalten.
Grundlagen der Informatik:

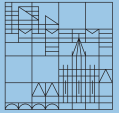
- Herold, Lurz & Wohlrab: *Grundlagen der Informatik* (lbs 830/h27, 3-8273-7305-2)



- Grumm & Sommer: *Einführung in die Informatik* (lbs 830/g95(7), 3-486-58115-7)
- Küchlin & Weber: *Einführung in die Informatik* (lbs 843/k92(3), 3-540-20958-1)

Algorithmen und Datenstrukturen

- Cormen: *Algorithmen - eine Einführung* (lbs 840/a53, ISBN 3-486-58262-8) oder das englische Original *Introduction to Algorithms* (lbs 830/c67(28), ISBN 0-262-03293-7)
- Sedgewick: *Algorithmen in Java* (Teil 1-4) (kid 112:n/s26-1/4, ISBN 3-8273-7072-8) oder das englische Original inklusive Graphalgorithmen: *Algorithms in Java, Part 1-4* (ISBN 0-201-36120-9), *Algorithms in Java, Part 5* (ISBN 0-201-36121-6)
- Ottmann & Widmayer: *Algorithmen und Datenstrukturen* (lbs 830/o99(4), ISBN 3-8274-1029-0)



11.4 Rechnersysteme und -netze (Computer Systems)

Modulverantwortlicher : Prof. Dr. Berthold (z.B. INF-11880-20122)

Credits : 5 SWS, 6 ECTS

Inhalte :

Die Vorlesung behandelt Grundlagen der Technischen Informatik wie Digitale Schaltungstechnik, Boolesche Algebra, Sequentielle Logik, Maschinensprache, Computerarchitektur, Assembler, Virtuelle Maschinen, Höhere Programmiersprachen, Compiler, Betriebssysteme und Netzwerktechnik.

Lernziele :

Absolventen haben ein grundlegendes Verständnis der kombinatorischen und sequentiellen Schaltungstechnik. Die grundlegenden Konzepte von Rechnerarchitekturen und Betriebssystemen (z.B. von Neumann) sind ebenso verstanden wie Techniken des Compilerbaus, der Virtuellen Maschinen und Assembler. Des Weiteren sollen Studierende Modelle der Netzwerktechnik (z.B. 5-Schichten-Modell) erörtern und verschiedene Protokolle (z.B. HTTP, SMTP, TCP, IP, ...) erläutern und in die besprochenen Modelle einordnen können.

Voraussetzung :

Keine

Arbeitsaufwand :

Präsenzstudium: 70 Stunden, Eigenstudium: 110 Stunden

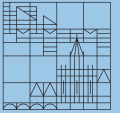
Prüfungsleistung :

- Studienleistung: 50% der Punkte aus den Quizzes
- Prüfungsleistung: Klausur von 120 Minuten Dauer, Teilnahmevoraussetzung ist das Absolvieren der Studienleistung.
- Die Note entspricht der Klausurnote

Literatur :

Rechnersysteme:

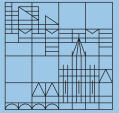
- *The Elements of Computing Systems: Building a Modern Computer from First Principles* von Noam Nisan und Shimon Schocken, 2008, MIT Press (ISBN: 978-0-262-14087-4, lbs 846/n48:a)
- David A. Patterson, John L. Hennessy: *Rechnerorganisation und -entwurf – Die Hardware / Software-Schnittstelle* (3. Auflage 2005). Englische Ausgabe: *Computer Organization and Design*. Morgan Kaufmann Publishers



- Andrew S. Tanenbaum: *Computerarchitektur, Strukturen - Konzepte - Grundlagen* (5. Auflage 2006), Pearson/Prentice Hall; Englische Ausgabe: *Structured Computer Organization*
- Viele andere Bücher, z.B. durch eine Suche auf <http://www.ub.uni-konstanz.de> nach "Betriebssysteme" oder "Operating Systems".

Netzwerke:

- Kurose/Ross. *Computer Networking – A Top-Down Approach Featuring the Internet*. Addison-Wesley 2001
- Walter Proebster. *Rechnernetze: Technik, Protokolle, Systeme, Anwendungen*. Oldenbourg 1998



11.5 Analysis I

Modulverantwortlicher : FB Mathematik (z.B. MAT-10056-20112)

Credits : 6 SWS, 9 ECTS

Inhalte :

Zentral sind Zahlen, Folgen, Reihen, metrische und normierte Räume, Stetigkeit, Differenzierbarkeit, Integration, Reihen von Funktionen

Lernziele :

Erwerb grundlegender Kenntnisse in den Grundlagen der Analysis, insbesondere in den Methoden der Differential- und Integralrechnung von Funktionen einer Veränderlichen

Voraussetzung :

Keine

Arbeitsaufwand :

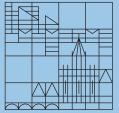
Präsenzstudium: 84 Stunden, Eigenstudium: 186 Stunden

Prüfungsleistung :

schriftliche Prüfung von 90 Minuten Dauer. Die Note ergibt sich aus der schriftlichen Prüfungsleistung.

Literatur :

R. Denk, R. Racke: *Kompendium der Analysis*, Band 1. Vieweg+Teubner (2011).



11.6 Schlüsselkompetenzen der Informatik (Key Competences for Undergraduate Students of Computer Science)

Modulverantwortlicher : Prof. Dr. Scholl (z.B. INF-10175-20121)

Credits : 2 SWS, 3 ECTS

Inhalte :

Geeignete Werkzeuge und Techniken nebenläufig zum Studium zu finden und zu erlernen ist zeitaufwendig und mühsam. Aufgrund der Auslastung in den ersten Semestern hatte dies oft zur Folge, dass Kompetenzen, wie das Erstellen naturwissenschaftlicher Dokumente mit LaTeX, der Umgang mit Source Code Management Tools, allgemeine Unix-Kenntnisse, das Erstellen von Visualisierungen, der Umgang mit Build Tools und Debuggern, etc. nicht allgemein als bekannt vorausgesetzt werden konnten.

In der Veranstaltung Schlüsselkompetenzen der Informatik werden daher bewährte Tools und Techniken vorgestellt und gemeinsam erprobt. Es werden Grundkenntnisse im Umgang mit den Werkzeugen eines Informatikers vermittelt und praktisch angewendet.

Ziel ist es, einen Überblick zu vermitteln, wie reichhaltig der Fundus an frei verfügbaren Programmen ist, welche Werkzeuge etabliert sind und wie diese erlernt werden können (Hilfe zur Selbsthilfe).

Der Kurs konnte bisher sehr interaktiv durchgeführt werden. Das heißt, dass wir Software on-the-fly auf Euren Laptops installieren (bitte den Rechner mitbringen!) und gemeinsam die ersten Schritte bei der Verwendung durchführen.

Am Anfang des Kurses ist auch Zeit, um Fragen der Studienorganisation, die gerade im ersten Semester noch mühsam sein kann, zu klären und allgemeine Abläufe an der Uni und am Fachbereich vorzustellen. Der Kurs ist in zwei Semester geteilt. Neben der Studienberatung für Erstsemester sind die wesentlichen inhaltlichen Schwerpunkte dieses Semesters das Erstellen naturwissenschaftlicher Dokumente mit LaTeX und das Arbeiten mit UNIX-Systemen.

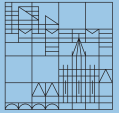
Lernziele :

Eine (praktische) Einführungsveranstaltung in die Studiengänge am FB

- Fachkompetenzen vermitteln
- Studieneinstieg erleichtern
- Praktische, fachliche Studienberatung leisten
- *Handwerkszeug* für vertiefende Veranstaltungen garantieren

Schwerpunkte:

- Werkzeuge, Techniken und Methoden kennen lernen, die im Studien- und später auch Berufsalltag einer Informatikerin benötigt werden.
- Einblick in den Studienverlauf, die organisatorischen Hürden und formalen Notwendigkeiten geben.



- Grundlagen und Voraussetzungen schaffen, die im weiteren Studienverlauf benötigt werden (und die dann auch von vertiefenden Veranstaltungen als bekannt vorausgesetzt werden).
- Den Einstieg in die Welt der Informatik und des Informatikstudiums am Fachbereich begleiten und erleichtern.

Voraussetzung :

Keine. Der Kurs ist explizit für Einsteiger, d.h. Erstsemester(!), in das Studium Information Engineering/Informatik am Fachbereich Informatik & Informationswissenschaft konzipiert.

Arbeitsaufwand :

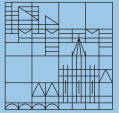
Präsenzstudium: 28 Stunden, Eigenstudium: 62 Stunden

Prüfungsleistung :

Kursbegleitende Aufgaben (unbenoteter Schein).

Literatur :

Wird während des Kurses bekannt gegeben.



12 Grundstudium - 2. Semester - Pflichtbereich

12.1 Konzepte der Programmierung (Programming Concepts)

Modulverantwortlicher : Prof. Dr. Deussen, Prof. Dr. Scholl (z.B. INF-12070-20131)

Credits : 5 SWS, 4 ECTS

Inhalte :

Das Modul Informatik 2 besteht aus der Vorlesung “Konzepte der Informatik“ und dem Programmierkurs 2. Kern des Moduls ist eine Einführung in deklarative und logische Programmierparadigmen. Im Unterschied zur imperativen Programmierung wird dabei durch den Programmierer nur vorgegeben, was berechnet werden soll, aber nicht wie genau die Berechnung durchgeführt wird. Die erste Hälfte der Vorlesung wird sich mit Haskell beschäftigen. Haskell ist ein Beispiel für eine reine funktionale Programmiersprache, die ohne Seiteneffekte (wie etwa Variablenzuweisungen) auskommt. In der zweiten Hälfte wird Prolog als Beispiel einer logischen Programmiersprache behandelt. Vorlesungsbegleitend gibt der “Programmierkurs 2“ eine praktische Einführung in diese beiden Programmiersprachen. Da Vorlesung und Programmierkurs inhaltlich eng verzahnt sind, werden die Übungen zu beiden Veranstaltungen zusammengelegt.

Lernziele :

Absolventen haben ein grundlegendes Verständnis von Programmierparadigmen und von deklarativer und logischer Programmierung. Sie sind in der Lage, selbständig kleinere Projekte in Haskell und Prolog zu definieren und zu implementieren.

Voraussetzung :

keine

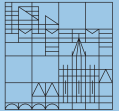
Arbeitsaufwand :

Insgesamt 120 Stunden

- 65 Stunden Präsenzstudium
- 55 Stunden Eigenstudium

Prüfungsleistung :

Leistungsnachweis: schriftliche Prüfung von 120 Minuten Dauer
erfolgreiche Teilnahme an den Übungen ist Voraussetzung für die Zulassung zur Prüfung
Note: die Note ergibt sich aus der Note der schriftlichen Prüfung



12.2 Programmierkurs 2 (Programming Course 2)

Modulverantwortlicher : Prof. Dr. Deussen, Prof. Dr. Scholl (z.B. INF-12120-20131)

Credits : 4 SWS, 5 ECTS

Inhalte :

Das Modul Informatik 2 besteht aus der Vorlesung *Konzepte der Informatik* und dem *Programmierkurs 2*.

Kern des Moduls ist eine Einführung in deklarative und logische Programmierparadigmen. Im Unterschied zur imperativen Programmierung wird dabei durch den Programmierer nur vorgegeben, was berechnet werden soll, aber nicht wie genau die Berechnung durchgeführt wird. Die erste Hälfte der Vorlesung wird sich mit Haskell beschäftigen. Haskell ist ein Beispiel für eine reine funktionale Programmiersprache, die ohne Seiteneffekte (wie etwa Variablenzuweisungen) auskommt. In der zweiten Hälfte wird Prolog als Beispiel einer logischen Programmiersprache behandelt. Vorlesungsbegleitend gibt der "Programmierkurs 2" eine praktische Einführung in diese beiden Programmiersprachen. Da Vorlesung und Programmierkurs inhaltlich eng verzahnt sind, werden die Übungen zu beiden Veranstaltungen zusammengelegt.

Lernziele :

Absolventen haben ein grundlegendes Verständnis von Programmierparadigmen und von deklarativer und logischer Programmierung. Sie sind in der Lage, selbständig kleinere Projekte in Haskell und Prolog zu definieren und zu implementieren

Voraussetzung :

Gleichzeitiger oder vorausgehender Besuch der Vorlesung "Konzepte der Programmierung"

Arbeitsaufwand :

150 Stunden insgesamt

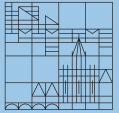
- 52 Stunden Präsenzstudium
- 98 Stunden Eigenstudium

Prüfungsleistung :

schriftliche Prüfung wie angegeben in "Konzepte der Programmierung"

Literatur :

wird im Kurs bekannt gegeben



12.3 Datenbanksysteme (Database Systems)

Modulverantwortlicher : Prof. Dr. Scholl (z.B. INF-12040-20131)

Credits : 6 SWS, 9 ECTS

Inhalte :

Die Veranstaltung vermittelt einen grundlegenden Überblick über Funktionalität, Architektur und Realisierungskonzepte von Datenbanksystemen als Grundlage für computergestützte Informationssysteme. Charakteristisch für Datenbanksysteme ist, dass Informationen gemäß irgendeinem Modell dargestellt, gespeichert und aufbewahrt werden, die mittels Operationen einer geeigneten Sprache abgefragt (wiedergewonnen) und manipuliert werden können. In dieser Veranstaltung werden sowohl die Modellierungs- als auch die Nutzungsaspekte von Datenbanksystemen vermittelt: z.B. Entity-Relationship- und Relationale Datenmodellierung, Relationale Entwurfstheorie und Normalformen, Datenbanksprachen (insbes. Algebra, Kalkül, SQL), ACID-Transaktionen. Die Lehrveranstaltung liefert Grundlagen für weiterführende Lehrveranstaltungen aus den Gebieten Datenbanken, Informationssysteme und Information Retrieval.

Lernziele :

Absolventen des Kurses haben grundlegendes Verständnis über den Aufbau und die Funktionsweise von Datenbanksystemen und deren Nutzung. Sie haben fundiertes Wissen über konzeptionelle Datenmodellierung mit Hilfe des Entity-Relationship-Modells und die Abbildung auf relationale Datenbankschemata. Sie können die grundlegenden Sprachkonstrukte von SQL mittels mathematisch präziser formaler Sprachen (Algebra, Kalkül) analysieren und können SQL-Anfragen und -Änderungsoperationen selbständig formulieren und anwenden. Sie haben die prinzipiellen Realisierungstechniken solcher deklarativer Sprachen kennen gelernt und können bestehende SQL-Anwendungen analysieren und bewerten. Sie sind in der Lage, grundlegende Informationssystem-Funktionalitäten selbständig zu realisieren. Die Funktionsweise und Abstraktionsmechanismen der transaktionsorientierten Verarbeitung sind ihnen bekannt, sie können Synchronisations- und Recovery-Probleme erkennen und grundsätzliche Lösungsmöglichkeiten aufzeigen.

Voraussetzung :

Entsprechend den Modulen Mathematik: Diskrete Strukturen (paralleler Besuch) und Konzepte der Informatik, elementare Programmierkenntnisse.

Arbeitsaufwand :

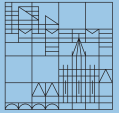
270 Stunden, davon ca. 90h Präsenz und 180h Selbststudium

Prüfungsleistung :

Prüfung: Klausur von 120min Dauer. Die erfolgreiche Teilnahme an den Übungen ist Voraussetzung für die Zulassung zur Klausur. Die Note ergibt sich aus der Klausurnote.

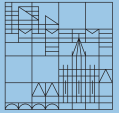
Literatur :

Es gibt eine große Auswahl an guten Standard-Lehrbüchern über Datenbanken, sowohl in



deutscher wie auch in englischer Sprache (siehe auch DB-Literatur). Die Veranstaltung folgt einem eigenen Aufbau. Als Einstieg kann dennoch bspw. dienen

- Kemper, A. und A. Eickler (2009). *Datenbanksysteme: Eine Einführung*. Oldenbourg, 7. Aufl.



12.4 Mathematik: Diskrete Strukturen (Mathematics: Discrete Structures)

Modulverantwortlicher : PD Dr. Kosub (z.B. INF-12090-20131)

Credits : 6 SWS, 9 ECTS

Inhalte :

- Algebraische Strukturen (Algebren, Gruppen, endliche Körper)
- Kombinatorik (Abzählprinzipien, Urnenmodelle, Binomialkoeffizienten, Permutationen, Partitionen)
- Graphentheorie (Graphen, Bäume und Wälder, Kreise, planare Graphen, Färbungen, Matchings)
- Rekursionen (Lineare Rekursionsgleichungen, erzeugende Funktionen, höhere Rekursionsgleichungen)
- Diskrete Wahrscheinlichkeitstheorie (Wahrscheinlichkeitsräume, bedingte Wahrscheinlichkeit, Unabhängigkeit, Zufallsvariablen, Gesetz der großen Zahlen, wichtige diskrete Verteilungen)

Lernziele :

Beherrschung der grundlegenden Konzepte der Diskreten Mathematik, wie sie für Informatik und Information Engineering benötigt und vorausgesetzt werden

Voraussetzung :

Kenntnisse entsprechend der Vorlesung Mathematische Grundlagen der Informatik

Arbeitsaufwand :

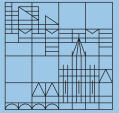
Präsenzstudium: 84 Stunden, Eigenstudium: 186 Stunden

Prüfungsleistung :

Klausur, 120 Minuten, sowie 50% der Punkte aus den wöchentlichen Übungen

Literatur :

- Angelika Steger. *Diskrete Strukturen 1. Kombinatorik, Graphentheorie, Algebra*. 2., Auflage, Springer-Verlag, Berlin, 2007.
- Thomas Schickinger, Angelika Steger. *Diskrete Strukturen 2. Wahrscheinlichkeitstheorie und Statistik*. 2., Auflage, Springer-Verlag, Berlin, 2007.



13 Grundstudium - 3. Semester - Pflichtbereich

13.1 Betriebssysteme und systemnahe Programmierung

Modulverantwortlicher : Prof. Dr. Waldvogel (z.B. INF-10835-20091)

Credits : 4 SWS, 6 ECTS

Inhalte :

Die Veranstaltung beinhaltet eine grundlegende Einführung in Konzepte zum Aufbau und Methoden zum Betrieb von Betriebssystemen. Die Vorlesung bietet einen theoretischen Einblick in die allgemeinen Konzepte und Problemstellungen. Wann immer konkrete Umsetzungen und Beispiele behandelt werden, wird Bezug auf das freie Betriebssystem GNU/Linux genommen. Der Besuch des Programmierkurses III - Systemnahe Programmierung wird als Parallelveranstaltung durch die Prüfungsordnung (BA Informatik) gefordert und durch die Lehrenden empfohlen.

Lernziele :

Die Verwendung, Spezifikation und Dokumentation der Systemprogrammierstelle (natives API, system call interface) von Betriebssystemen findet (fast immer) mittels der Programmiersprache C statt. Eine grundlegende Einführung in diese Sprache ist daher Bestandteil der ersten Übungsstunden der Veranstaltung. Die Vorlesung kann als Grundlagenveranstaltung für zahlreiche weiterführende Veranstaltungen (siehe unten) verstanden werden und soll Studenten befähigen, Bachelor-/Masterprojekte an system- und/oder hardwarenahen Forschungsprojekten am Fachbereich annehmen zu können. Weiterführende Veranstaltungen: Rechnernetze, Computergrafik, Architektur von Datenbankmanagementsystemen, Systemprogrammierung ...

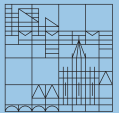
Voraussetzung : Rechnersysteme

Arbeitsaufwand : Präsenzstudium: 56 Stunden, Eigenstudium: 124 Stunden

Prüfungsleistung : Klausur

Literatur :

- William Stallings, *Operating Systems, Internals and design principles*, 6th / 7th ed. <http://williamstallings.com/OS/OS6e.html>, <http://williamstallings.com/OperatingSystems/>
- Stevens, W. Richard, *Advanced programming in the UNIX environment*, Addison-Wesley, 2001, 744 S., ISBN 0-201-56317-7
- Eduard Glatz, *Betriebssysteme - Grundlagen, Konzepte, Systemprogrammierung*, dpunkt.verlag, Heidelberg, ISBN 3-89864-355-7, lbs 843/g51, kid 270:n/g51
- Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, *Operating Systems Concepts*, Wiley & Sons, Hoboken, NJ, 2005, lbs 843/s45(7), kid 270/p29(7)



13.2 Programmierkurs 3 - Systemnahe Programmierung (Systems Programming)

Modulverantwortlicher : Prof. Dr. Waldvogel (z.B. INF-11850-20102)

Credits : 4 SWS, 5 ECTS

Inhalte :

Systems programming reviews operating systems concepts from a practical perspective using Unix/Linux. Students are expected to visit the accompanying Operating Systems lecture in parallel or have previous knowledge. The lecture is divided into three parts:

1. The C Programming Language
2. Linux System Programming
3. Linux Kernel Development

C is the programming language of Unix operating systems. A thorough discussion of the language with accompanying weekly assignments lays foundation for the following parts.

Lernziele :

Knowing the services provided by the Linux kernel (system calls) and the C library forms the basis for advanced programming in the Linux/Unix environment. Topics include File, Buffered, and Advanced I/O (such as scatter/gather, poll, memory mapped I/O), Process Management, File and Directory Management, Memory Management, Signals, and Time facilities. Linux Kernel Development provides an insight on how to read and enhance kernel code. Topics include the development of Linux kernel modules and simple device drivers. Aspects like kernel threads and basic kernel structures are discussed and reviewed.

Voraussetzung :

Rechnersysteme (Computer Architecture, Assembler), (Operating Systems), basic (user) knowledge of Unix systems

Arbeitsaufwand :

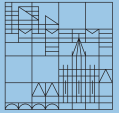
Präsenzstudium: 56 Stunden, Eigenstudium: 94 Stunden

Prüfungsleistung :

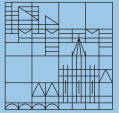
Klausur am Ende des Semesters (Termin wird noch bekannt gegeben).
Zulassung zur Klausur: Jedes Übungsblatt muss zu mindestens 50% gelöst worden sein.

Literatur :

- Kernighan, Brian W. / Ritchie, Dennis M., *The C Programming Language*, 274 pages, April 1988, 2nd ed., Prentice Hall, ISBN-10: 0131103628, ISBN-13: 978-0131103627
- Love, Robert, *Linux System Programming, Talking directly to the Kernel and C library*, 368 pages, October 2007, 1st ed., O'Reilly ISBN-10: 0596009585, ISBN-13: 978-0596009588
- Stevens, W. Richard / Rago, Stephen A., *Advanced Programming in the UNIX Environment*, 927 pages, February 2008, 2nd revised ed., Addison-Wesley Professional, ISBN-10: 0321525949, ISBN-13: 978-0321525949



- Love, Robert, *Linux Kernel Development, A thorough guide to the design and implementation of the Linux kernel*, 440 pages, Juli 2010, 3rd revised ed., Addison-Wesley, ISBN-10: 0672329468, ISBN-13: 978-0672329463



13.3 Software Engineering (Software Engineering)

Modulverantwortlicher : Prof. Dr. Leue (z.B. INF-11890-20122)

Credits : 5 SWS, 5 ECTS

Inhalte :

Die Veranstaltung Software Engineering führt in Verfahren, Methoden und Werkzeuge zum ingenieurmäßigen Entwurf von Softwaresystemen ein. Sie beschäftigt sich mit Software-Prozessmodellen, dem objektorientierten Entwurf von Software und ihrer Architektur, der Spezifikation, Verifikation und dem Testen von Software, der Planung und Durchführung von Softwareprojekten und der quantitativen Bewertung von Software-Artefakten. Gleichzeitig bereitet sie die TeilnehmerInnen auf die prozessorientierte Durchführung von Softwareprojekten vor. Die Veranstaltung Software Engineering bildet gemeinsam mit der Veranstaltung Software-Projekt ein Modul.

Lernziele :

Die TeilnehmerInnen werden in die Lage versetzt, Software-Entwicklungsmethoden zu bewerten, ihren Einsatz zu strukturieren, und selber Software-Entwicklungsprojekte zu leiten.

Voraussetzung :

Erfolgreiche Teilnahme an den Lehrveranstaltungen Modul Informatik I, Modul Informatik II, Diskrete Strukturen

Arbeitsaufwand :

Präsenzstudium: 70 Stunden, Eigenstudium: 80 Stunden

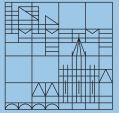
Prüfungsleistung :

Klausur, 90 Minuten, plus Studienleistungen. Details werden während der Veranstaltung bekannt gegeben.

Literatur :

- Ian Sommerville. *Software Engineering*. 9th edition. Pearson Education Ltd. 2010.
- Bernd Brügge, Allen H. Dutoit. *Object-Oriented Software Engineering. Using UML, Patterns and Java*. 3rd edition, Prentice-Hall, 2009.

Weitere Literatur wird während der Veranstaltung bekannt gegeben.



13.4 Algorithmen und Datenstrukturen (Algorithms and Data Structures)

Modulverantwortlicher : Prof. Dr. Brandes (z.B. INF-20250-20122)

Credits : 6 SWS, 9 ECTS

Inhalte :

In der Vorlesung werden Standardalgorithmen und Grund legende Datenstrukturen behandelt. Darstellungsformen und Spezifikation von Algorithmen, elementare und höhere Datenstrukturen, Suchbäume, Hash-Tabellen, rekursive Algorithmen, Algorithmen zum Suchen und Sortieren, Grund legende Graphenalgorithmen und Zeichenkettenalgorithmen. In theoretischen Übungen wird der Vorlesungsstoff vertieft, in praktischen Übungen werden Algorithmen und Datenstrukturen in Java implementiert.

Arbeitsaufwand :

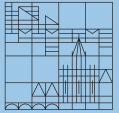
Präsenzstudium: 84 Stunden, Eigenstudium: 186 Stunden

Prüfungsleistung :

Teilnahme an den Übungen (50% der Punkte), Klausur (120min)

Literatur :

- N. Blum: *Algorithmen und Datenstrukturen*. Oldenbourg, 2004
- T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein: *Algorithmen - Eine Einführung*. Oldenbourg, 2007 (2. Aufl.)
- T. Ottmann, P. Widmayer: *Algorithmen und Datenstrukturen*. Spektrum Akademischer Verlag, 2002 (4. Aufl.)
- U. Schöning: *Algorithmik*. Spektrum Akademischer Verlag, 2001
- M.A. Weiss: *Data Structures and Algorithm Analysis in Java*. Pearson, 2007 (2nd ed.)



13.5 Lineare Algebra I (Linear Algebra I)

Modulverantwortlicher : FB Mathematik (z.B. MAT-10066-20112)

Credits : 6 SWS, 9 ECTS

Inhalte :

Algebraische Grundlagen, Vektorräume, Lineare Abbildungen und ihre Darstellbarkeit durch Matrizen, Dual und Bidual, der Transponierte Operator, Determinanten.

Algebraic foundations, vector spaces, linear maps between vector spaces and their representability by matrices, Dual and Double dual space, the transpose of a linear operator, Determinants.

Lernziele :

Die Studenten sollen instand gesetzt werden, selbständig und souverän mit abstrakten Konzepten umzugehen, einfache Beweise eigenständig zu führen, mathematisch exakt darzustellen und vorgelegte Beweisversuche kritisch auf ihre Korrektheit zu überprüfen.

The students are to be enabled to independently handle abstract notions, find easy proofs, present those in accordance with the standards of mathematical rigor and to critically check proof attempts for their correctness.

Voraussetzung :

Grundkenntnisse im Umgang mit logischen Zeichen, Mengen, Funktionen und einfachen Beweisformen (direkt/indirekt/induktiv), wie sie z.B. im Vorkurs Mathematik vor Semesterbeginn vermittelt werden.

Basic skills in handling logical symbols, sets, functions and proofs (direct/indirect/inductive) as taught in the preparatory class for mathematics prior to the start of the semester.

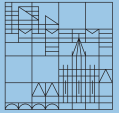
Arbeitsaufwand :

Präsenzstudium: 84 Stunden, Eigenstudium: 186 Stunden

Prüfungsleistung :

Es gibt eine Klausur Anfang Februar und eine zweite Anfang März. Die Endnote ist die bessere der folgenden beiden Noten: $1/5 \cdot \text{Note 1. Klausur} + 4/5 \cdot \text{Note 2. Klausur}$ bzw. Note 2. Klausur. Außerdem sind zum Bestehen mindestens 50% der Punkte bei Bearbeitung der Übungsblätter zu erreichen. Ist das der Fall und ist die Endnote mindestens 4.0, ist das Modul bestanden. Der Leistungsnachweis erfolgt in Form eines benoteten Scheins.

There will be a first exam early in february and a second exam early in march. The final grade is the better one of the following two grades: $1/5 \cdot \text{first exam grade} + 4/5 \cdot \text{second exam grade}$ and second exam grade. For passing, it is also required to score at least 50% on the exercise sheets. If this is the case and the final grade is at least 4.0, the student has passed the module and receives a graded certificate.



13.6 Analyse & Visualisierung (Analysis & Visualization)

Modulverantwortlicher : Prof. Dr. Keim (z.B. INF-11710-20122)

Credits : 6 SWS, 9 ECTS

Anmerkung :

Ist Grundstudiumsveranstaltung für Bachelor Informatik. Kann für Bachelor/Master Information Engineering im Vertiefungsstudium gehört werden.

Inhalte :

- Grundlegende Methoden des Data Mining
- Datenaufbereitung und Datenvorverarbeitung
- Gebräuchliche Analysealgorithmen und -methoden
- Grundlagen der Informationsvisualisierung
- Menschliche Wahrnehmung und Visualisierungsdesign
- Visualisierungsmöglichkeiten verschiedener Daten

Lernziele :

Die Studierenden erhalten grundlegende theoretische Kenntnisse und machen erste praktische Erfahrungen im Bereich der Datenanalyse. Sie sind in der Lage, Voraussetzungen und Parameter für die Anwendung gängiger verschiedener Analysealgorithmen zu benennen, diese praktisch anzuwenden und die erreichten Ergebnisse eigenständig zu beurteilen. Im Bereich der Visualisierung lernen sie, ausgehend von verschiedenen Datentypen, unterschiedliche geeignete visuelle Abbildungen für diese kennen. Anhand von Beispielen lernen die Studenten die qualitative Analyse der Visualisierungsgüte und können Designentscheidungen hinsichtlich der menschlichen Wahrnehmung autonom beurteilen. Dadurch können die Teilnehmer selbständig Visualisierungen entwickeln und analysieren.

Voraussetzung :

Modul Informatik 1, Mathematische Grundlagen der Informatik

Arbeitsaufwand :

SWS: 6 Vorlesung / Übung Insgesamt 270 Stunden, verteilt auf

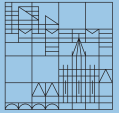
- 84 Stunden Präsenzstudium
- 186 Stunden Eigenstudium

Prüfungsleistung :

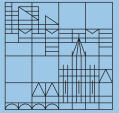
Klausur sowie erfolgreiches Absolvieren von mind. 50% der Übungsaufgaben

Literatur :

- Han J., Kamber M., *Data Mining: Concepts and Techniques*, 2006, Morgan Kaufmann Publishers, Second Edition



- Berthold M., Borgelt C., Höppner F., Klawonn F., *Guide to Intelligent Data Analysis: How to Intelligently Make Sense of Real Data: Making Practical Sense of Real Data* (Texts in Computer Science), 2010 Springer
- Ester M., Sander J., *Knowledge Discovery in Databases. Techniken und Anwendungen*, 2000 Springer Verlag
- Hand D.J., Mannila H., Smyth P., *Principles of Data Mining*, 2001, MIT Press
- Spence R., *Information Visualization*, 2007, ACM Press Books, Second Edition
- Ward M. and Grinstein, G. and Keim D. A., *Interactive Data Visualization: Foundations, Techniques, and Application*, 2010, A.K. Peters, Ltd, ISBN: 978-1-56881-473-5, <http://www.idvbook.com>



14 Grundstudium - 4. Semester - Pflichtbereich

14.1 Software-Projekt (Software Project)

Modulverantwortlicher : PD Dr. Kosub (z.B. INF-12160-20121)

Credits : 4 SWS, 6 ECTS

Inhalte :

Die Studierenden entwickeln in Projektgruppen gemäß einem vorgegebenen Prozessmodell ein Softwaresystem mittlerer Komplexität. Dabei erstellen sie eine Anforderungsspezifikation (Abgabe 1) und einen Softwareentwurf, der aus einem architekturellen Entwurf (Abgabe 2) und einem Detailentwurf (Abgabe 3) besteht. Die Abgabe 4 besteht aus der Implementierung des entworfenen Systems.

Lernziele :

Die Studierenden werden in die Lage versetzt, in strukturierter und kollaborativer Art und Weise eine komplexe Aufgabenstellung zu analysieren, die Analyse in einen Softwareentwurf umzusetzen, diesen zu implementieren und seine Übereinstimmung mit der Anforderungsanalyse durch geeignetes Testen nachzuweisen. Sie erkennen damit, dass die Entwicklung komplexer Softwaresysteme eine prozessgesteuerte und gruppenorientierte Arbeitsweise voraussetzt. Sie werden dadurch befähigt, in ihrem späteren Berufsleben an komplexen kollaborativen Softwareentwicklungsprojekten mitzuwirken und diese zu leiten.

Voraussetzung :

Kenntnisse entsprechend der Veranstaltungen *Konzepte der Informatik* und *Konzepte der Programmierung und Algorithmen und Datenstrukturen*. Erfolgreiche Teilnahme an der Veranstaltung *Software Engineering*, mit der gemeinsam dieses Praktikum das Modul *Software Engineering* darstellt. Die Veranstaltung *Software Engineering* legt die notwendigen methodischen Grundlagen, ohne die dieses Projekt nicht erfolgreich absolviert werden kann.

Arbeitsaufwand :

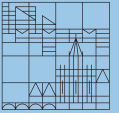
6 ECTS Credits, entspricht 180 Stunden. Davon: Präsenzzeit in geringem Umfang (ca. 3 Tutorien im Semester), der Rest entspricht Eigenstudium (Konsultationszeit mit Gruppenbetreuern, Erstellung der Projektabgaben, Implementierung).

Prüfungsleistung :

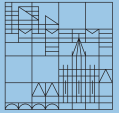
Studienleistung: Projektabgaben (Deliverables) und Implementierung des Systems
Prüfung: Präsentation der Projektabgaben, Präsentation der lauffähigen Implementierung des Systems, anschließende mündliche Prüfung
Note: gewichtetes Mittel der Teilleistungen

Literatur :

- Bernd Brügge, Allen H. Dutoit, *Object Oriented Software Engineering Using UML, Patterns, and Java: International Version*, Prentice Hall International; 3rd revised edition. August 2009.



- Helmut Balzert, *Lehrbuch der Software-Technik 1/2*, Spektrum Akademischer Verlag; 2. Auflage. Dez. 2000.



14.2 Theoretische Grundlagen der Informatik (Theoretical Computer Science)

Modulverantwortlicher : Prof. Dr. Saupe (z.B. INF-2101-20121)

Credits : 6 SWS, 9 ECTS

Inhalte :

Die Vorlesung gibt eine Einführung in die theoretischen Grundlagen der Informatik. Folgende Themen werden u.a. behandelt:

- Formale Sprachen und Automatentheorie Chomsky-Hierarchie (reguläre, kontextfreie, Typ0-Sprachen, reguläre Ausdrücke) Grammatiken (Typen, Eindeutigkeit, Abgeschlossenheit) Automatenmodelle (endliche Automaten, Kellerautomaten, Turingmaschinen)
- Entscheidbarkeit und Berechenbarkeit Entscheidbarkeit, Aufzählbarkeit Universelle Turingmaschine, Diagonalisierung, Halteproblem Berechenbarkeit, μ -rekursive Funktionen, Church/Turing-These
- Komplexitätstheorie Optimierungs- und Entscheidungsprobleme Codierung Klassen P und NP, NP-Vollständigkeit Parametrisierte Komplexität

Lernziele :

Der/die Studierende

- besitzt einen Einblick in die Grundlagen der Theoretischen Informatik und beherrscht deren Berechnungsmodelle und Beweistechniken,
- hat die Fähigkeit, die Standardkonstruktionen aus dem Bereich endlicher Automaten, regulärer Ausdrücke und Grammatiken auszuführen,
- hat ein Verständnis für die Unterscheidung von Berechenbarkeit und Unberechenbarkeit sowie ein Grundverständnis des Begriffs der NP-Vollständigkeit und seiner Motivation.

Voraussetzung :

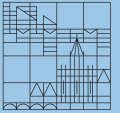
Keine. Es wird empfohlen, die Vorlesung Algorithmen und Datenstrukturen zuvor gehört zu haben.

Arbeitsaufwand :

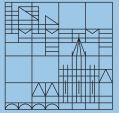
- ECTS: 9 Punkte
- SWS: 4h Vorlesung, 2h Übung.
- Insgesamt: 270 Stunden, davon 78 Stunden Präsenzstudium.

Prüfungsleistung : Klausur

Literatur :



- U. Schöning, *Theoretische Informatik - kurzgefasst*, Spektrum Akademischer Verlag, 5. Auflage, 2008.
- J. E. Hopcroft, R. Motwani, J. D. Ullman, *Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie*, Pearson Studium, 2. Auflage, 2002.
- I. Wegener, *Theoretische Informatik - eine algorithmische Einführung*, B. G. Teubner Verlag, 3. Auflage, 2011.



14.3 Computergrafik und interaktive Systeme (Computer Graphics and interactive Systems)

Modulverantwortlicher : Prof. Dr. Reiterer, Prof. Dr. Deussen (z.B. INF-12020-20131)

Credits : 6 SWS, 9 ECTS

Inhalte :

Die Veranstaltung besteht aus zwei Teilen: Im Teil Computergrafik wird eine Einführung in die interaktive Computergrafik unter OpenGL gegeben. Die Studierenden lernen den Weg von den Eingabedaten (geometrischen Beschreibungen der Objekte) bis hin zum Pixel des Ausgabebildes kennen

- Daten-Vorverarbeitung (Transformation, Projektion, Clipping)
- Rasterisierung (Scanline-Verfahren, Tiefenpuffer)
- Schattierungsmethoden (Gourand shading, Phong shading)
- lokale vs. globale Beleuchtungsverfahren
- Raytracing, Radiosity und bildbasiertes Rendering
- Texturierung

Im Teil Interaktive Systeme wird ein umfassender Überblick über die Ziele und Forschungsfragen der Disziplin Mensch-Computer-Interaktion gegeben. Die Studierenden erhalten ein Grundverständnis, wie man interaktive Systeme unter besonderer Berücksichtigung der Benutzeranforderungen entwickeln kann.

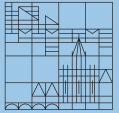
- Grundlagen der menschlichen Wahrnehmung
- Methoden und Techniken des User-Interface Designs
- Design von gebrauchstauglichen Produkten, deren Benutzung auch Spaß bereiten soll
- grundlegende Gestaltungsprinzipien, Methoden und Werkzeuge werden vorgestellt und in praktischen Übungen vertieft
- Techniken zur Evaluation von Benutzerschnittstellen

Lernziele :

Absolventen des Kurses haben grundlegendes Verständnis über den Aufbau graphisch- interaktiver Systeme und deren Realisierung mit OpenGL. Sie haben fundiertes Wissen über die Rasterisierungs-Pipeline und können sie in unterschiedlichen Kontexten anwenden und einsetzen. Sie haben grundlegende Interaktionskonzepte kennengelernt und können existierende interaktive Systeme analysieren und bewerten. Sie sind in der Lage, grundlegende Interaktionsdesigns selbständig zu realisieren.

Voraussetzung :

Entsprechend den Modulen Mathematische Grundlagen des Information Engineering und Methoden der Praktischen Informatik, elementare Programmierkenntnisse.



Arbeitsaufwand :

180 Stunden insgesamt

- 52 Stunden Präsenzstudium
- 128 Stunden Eigenstudium

Prüfungsleistung :

Prüfung: Klausur von 120min Dauer. Die erfolgreiche Teilnahme an den Übungen ist Voraussetzung für die Zulassung zur Klausur. Die Note ergibt sich aus der Klausurnote.

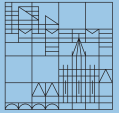
Literatur :

Teil Computergrafik:

- Edward Angel: *Interactive Computer graphics, a top-down approach with OpenGL* (in der Bibliothek mehrfach vorhanden - Die Vorlesungsfolien reichen nicht als Vorbereitung zur Prüfung!)

Teil Interaktive Systeme:

- Preim, B., Dachsett, R.: *Interaktive Systeme - Band 1*, (2. Auflage), Springer Verlag, 2010
- Richter, M., Flückiger, M. D.: *Usability Engineering kompakt: Benutzbare Software entwickeln*, (2.Auflage), Springer Verlag, 2010
- Preece J., Rogers Y., Sharp H.: *Interaction Design*, (3rd edition), John Wiley & Sons, 2011



14.4 Statistik I

Anmerkung

Alle an der Universität Konstanz angebotenen Veranstaltungen zu Statistik I werden für dieses Modul anerkannt.

Modulverantwortlicher :

z.B., FB Psychologie (z.B. PSY-10094-20122)

Credits :

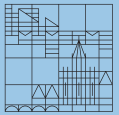
4 SWS, 6 ECTS

Lernziele :

Die Studierenden kennen die Grundlagen und Grundbegriffe der Methodologie und Methodik. Die Studierenden können einen Zusammenhang herstellen zwischen verschiedenen Spezialgebieten der psychologischen Methodenlehre untereinander sowie mit substanzwissenschaftlichen Inhalten der Psychologie. Die Studierenden verbessern ihr methodisches Denken und ihre Argumentationsfähigkeit. Die Studierenden können Strategien zur Aufgabenlösung entwickeln und anwenden. Die Studierenden können besser planen und Probleme lösen. Die Studierenden orientieren sich auf Anwendungen. Die Studierenden sind leistungsbereit, frustrationstolerant und belastbar. Präsenzstudium: 56 Stunden, Eigenstudium: 124 Stunden

Prüfungsleistung :

Klausur



15 Vertiefungsstudium

15.1 Data Management in the Cloud (Data Management in the Cloud)

Modulverantwortlicher : Jun.-Prof. Dr. Grossniklaus (z.B. INF-12820-20131)

Credits : 3 SWS, 5 ECTS

Inhalte :

Cloud computing has recently seen a lot of attention from research and industry for applications that can be parallelized on shared-nothing architectures and have a need for elastic scalability. As a consequence, new data management requirements have emerged with multiple solutions to address them. This course will look at the principles behind data management in the cloud as well as discuss actual cloud data management systems that are currently in use or being developed. The topics covered in the course range from novel data processing paradigms (MapReduce, PigLatin, Hive), to commercial cloud data management platforms (Google AppEngine, Microsoft Windows Azure, Amazon Web Services) and open-source NoSQL databases (Cassandra, MongoDB, Neo4j). The world of cloud data management is currently very diverse and heterogeneous. Therefore, our course will also report on efforts to classify, compare and benchmark the various approaches and systems. Students in this course will gain broad knowledge about the current state of the art in cloud data management and, through a course project, practical experience with a specific system.

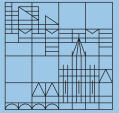
Lernziele :

The syllabus of this course aims at empowering students to follow and understand current trends in cloud computing and how they impact data management systems. Based on the acquired knowledge students will be capable to critically assess and differentiate between novel and emerging cloud data management systems. In particular, the learning content of this course will improve the students' ability to relate the requirements and characteristics of applications to the features and functionality of cloud data management systems. In an increasingly heterogeneous market of data management systems, the skill to make these informed choices is a key competence of graduates in computer science.

Voraussetzung :

The following skills, knowledge and courses are **mandatory** prerequisites to attend and successfully complete this course.

- **Basics of database systems:** database design, query languages, database application programming, etc. (INF-12040 or equivalent)
- **Principles of database systems:** relational model, relational algebra, normal forms, etc. (INF-12040 or equivalent)
- **Computer systems:** computer architecture, operating systems, networks, etc. (INF-11740, INF-11880, or equivalent)
- **System programming:** students must have the ability to program in a language appropriate for system implementation, such as C/C++, C# or Java. (INF-11930 or equivalent)



- **Key competences:** Subversion, LaTeX, etc. (INF-10175 or equivalent)

Arbeitsaufwand :

150 hours, of which ca. 50 hours are spent in class and another 100 hours are spent on assignments and the course project. Since cloud data management is a relatively new field, self-study and practical exercises are an integral part of this course. Therefore, these activities will take up 66% of the amount of work and will have an influence of 50% on the final grade.

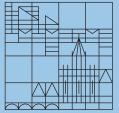
Prüfungsleistung :

Depending on the number of participants, there will be an oral exam (up to 30 students) or a written exam (more than 30 students). The type of exam will be announced in the first week of the course. Reading assignments and a practical project are an integral part of the lecture and will contribute 50% of the final grade. The reading assignments will be weighted as 20% and the project as 30%.

Literatur :

The following study material and literature is available and relevant for this course.

- **Course slides:** Copies of the slide decks used in the lecture as well as hand-outs for the exercises and reading assignments that accompany the course will be available online.
- **Reading list:** Publications referenced in the course and further readings will be published on the course website in the form of a reading list.



15.2 Geographic Information Systems (GIS) (Geographic Information Systems (GIS))

Modulverantwortlicher : Prof. Dr. Keim (z.B. INF-10615-20131)

Credits : 4 SWS, 6 ECTS

Inhalte :

Einführung in

- Geographische Informationssysteme
- Abbildung auf das relationale Datenmodell
- Räumliche Indexstrukturen
- Räumliche Anfragebearbeitung
- Algorithmen der Computer-Geometrie
- Rasterdaten
- Kombination von Raster- und Vektordaten
- Methoden der visuellen Analyse von Geo-Daten

Lernziele :

Die Studierenden erwerben die Fähigkeit, mit geographiebezogenen Daten umzugehen und relevante Verfahren der Informatik, speziell auf den Gebieten der Räumlichen Indexstrukturen und Computer-Geometrie, effizient einzusetzen. Nach erfolgreicher Teilnahme an der Veranstaltung beherrschen die Studierenden die Grundkonzepte geographischer Informationsverarbeitung, d.h. sie erwerben Kenntnisse der relevanten Begriffe, Anwendungsgebiete, Datenstrukturen, Datengrundlagen und Analyseoperationen, um komplexe GIS-Szenarien auf diese zurück zu führen. Sie sind in der Lage, bestehende GIS-Systeme hinsichtlich ihrer Leistungsfähigkeit zu beurteilen und anwendungsbezogen GIS-Systeme selbst zu konzipieren und zu realisieren.

Voraussetzung :

Grundvorlesungen des Bachelor Studienganges

Zusätzlich empfohlen: Information Visualization Data Mining

Arbeitsaufwand :

Insgesamt 180 Stunden (90 Stunden Vorlesung und 90 Stunden Übung), verteilt auf 56 Stunden Präsenzstudium und 124 Stunden Eigenstudium

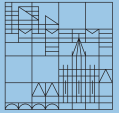
Prüfungsleistung :

Je nach Teilnehmer/innen/zahl mündlich (Einzelprüfung von 20 Minuten Dauer) oder schriftlich (Klausur von 60 Minuten Dauer); Projektnote fließt zu 1/3 in die Endnote ein.

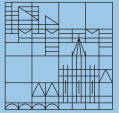
Literatur :

Grundlage ist ein (Folien-)Skript. Weitere Literaturempfehlungen werden zu Beginn der Vorlesung bekanntgegeben und zu jedem Vorlesungsabschnitt bekannt gegeben.

Allgemeine Literatur:



- Natalia Andrienko, Gennady Andrienko, *Exploratory analysis of spatial and temporal data: A systematic approach*, 2006, Springer
- J. Dykes, A.M. MacEachren, M.-J. Kraak, *Exploring geovisualization*, 2005, Elsevier
- Harvey J. Miller (Hrsg.), *Geographic data mining and knowledge discovery*, 2009, CRC Press
- Paul A. Longley, Michael F. Goodchild, David J. Maguire, David W. Rhind, *Geographic Information Systems & Science*, 2011, John Wiley & Sons, Inc.



15.3 Computational Methods for Document Analysis (Computational Methods for Document Analysis)

Modulverantwortlicher : Prof. Dr. Keim (z.B. INF-10585-20122)

Credits : 3 SWS, 5 ECTS

Inhalte :

- Natural Language Processing
- Information Retrieval
- Structure Analysis and Information Extraction
- Text Data Mining
- Text Visualization

Lernziele :

Die Studenten können die Problemstellungen bei der Dokumentenanalyse sowie die grundlegenden Algorithmen zu deren Lösung benennen. Sie sind in der Lage die Algorithmen auf spezielle Problemstellungen im Bereich der Dokumentenanalyse zu übertragen und ihre Lösung zu bewerten.

Voraussetzung :

Grundvorlesungen des Bachelor Studienganges. Von Vorteil ist Vorwissen in Data Mining und Informationsvisualisierung. Programmierkenntnisse in Java werden für das Bearbeiten der praktischen Übungen vorausgesetzt.

Arbeitsaufwand :

Insgesamt 150 Stunden, verteilt auf 42 Stunden Präsenzstudium und 108 Stunden Eigenstudium

Prüfungsleistung :

Bearbeitung der Übungen ist Zulassungsvoraussetzung zur Prüfung.

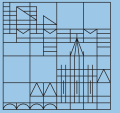
Prüfung: je nach Teilnehmer/innen/zahl mündlich (Einzelprüfung von 20 Minuten Dauer) oder schriftlich (Klausur von 120 Minuten Dauer).

Note: Die Note ergibt sich aus der Note der Prüfung.

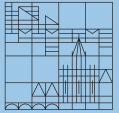
Literatur :

Keine vorausgesetzt, empfohlen werden:

- G. Heyer, U. Quasthoff, T. Wittig. *Text Mining: Wissensrohstoff Text – Konzepte, Algorithmen, Ergebnisse*. W3L-Verlag. 2006.
- C. D. Manning, P. Raghavan, H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press. 2008.
- Ruslan Mitkov. *The Oxford Handbook of Computational Linguistics*. Oxford University Press. 2005.



- Thomas Payne. *Exploring Language Structure: A Student's Guide*. Cambridge University Press. 2006.
- M. Ward, G. G. Grinstein, D. Keim. *Interactive Data Visualization: Foundations, Techniques, and Applications*. Taylor & Francis Ltd. 2010.



15.4 Applied Visual Analytics (Applied Visual Analytics)

Modulverantwortlicher : Prof. Dr. Keim (z.B. INF-10795-20131)

Credits : 2 SWS, 4 ECTS

Inhalte :

- Angewandte Methoden der Informationsvisualisierung
- Angewandte Methoden des Data Mining
- Interaktive Verknüpfung von automatischen und visuellen Methoden
- Anwendung und Anpassung moderner Forschungsmethoden auf wichtige Anwendungsgebiete, z.B. Exploration von Social Web Daten, oder Sicherheitsanwendungen (Epidemieerkennung, Kriminalitätsbekämpfung und innere Sicherheit)
- Umgang mit und Exploration von Daten wie Geokoordinaten, Gensequenzen, Textdokumenten, Daten aus Sozialen Netzwerken oder Internetdatenverkehr, jeweils im Zusammenhang mit zeitlichen Verläufen
- Teilnahme an einem internationalen Wettbewerb, z.B. der IEEE VAST Challenge

Lernziele :

Inhaltliche und vor allem methodische Kompetenzen im Bereich der Visuellen Datenanalyse:

- Projektplanung: Entwicklung von Strategien zur effizienten und effektiven Erschließung großer Datenquellen im Hinblick auf unscharfe Fragestellungen und zur Generierung von Hypothesen und Empfehlungen unter Unsicherheit.
- Schnittstelle Forschung-Praxis: Systematische Anwendung moderner Forschungsmethoden auf tatsächliche Probleme.
- Systematische Evaluierung verschiedener Programme und Werkzeuge zur visuellen Datenanalyse im Bezug auf ihre Praxistauglichkeit.
- Präsentation: Vermittlung von Analyseergebnissen an Entscheidungsträger und nachvollziehbare Dokumentation von Analyseverfahren. Die Studierenden besitzen vertiefte theoretische Kenntnisse und beherrschen das eigenständige Beurteilen der Vor- und Nachteile bestimmter Methoden in der Anwendung.

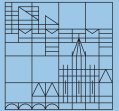
Voraussetzung :

Kenntnisse im Bereich Informationsvisualisierung und Data Mining oder paralleler Besuch entsprechender Vorlesungen erwünscht.

Arbeitsaufwand :

2 SWS Vorlesung/Seminar

Insgesamt 120 Stunden, verteilt auf 28 Stunden Präsenzstudium und 92 Stunden Eigenstudium

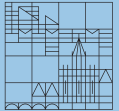


Prüfungsleistung :

Voraussetzung ist das Erstellen einer Lösung für die gegebene Analyse- und Visualisierungsaufgabenstellung. Die Aufgabenstellung kann von einem Wettbewerb aus dem Bereich der visuellen Datenanalyse (z.B. der IEEE VAST Challenge) stammen. Die Note ergibt sich aus der mündlichen Abschlusspräsentation und -prüfung über die Lösung.

Literatur :

- James J. Thomas and Kristin A. Cook. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Ctr, 2005.
- Matthew Ward and Georges Grinstein and Daniel A. Keim, *Interactive Data Visualization: Foundations, Techniques, and Application*, 2010, A.K. Peters, Ltd, ISBN: 978-1-56881-473-5, <http://www.idvbook.com>.
- Daniel Keim, Jörn Kohlhammer, Geoffrey Ellis, and Florian Mansmann (Editors): *Mastering the Information Age - Solving Problems with Visual Analytics*. E-Book verfügbar unter: <http://www.vismaster.eu/>
- IEEE VAST Challenge: <http://www.vacommunity.org/VAST+Challenge>



15.5 Neue Formen der Mensch-Computer-Interaktion: Blended Interaction (New Directions in Human-Computer Interaction: Blended Interaction)

Modulverantwortlicher : Prof. Dr. Reiterer (z.B. INF-11840-20122)

Credits : 6 SWS, 9 ECTS

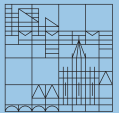
Inhalte :

Der Kurs “Blended Interaction - Neue Formen der Mensch-Computer-Interaktion“ gibt einen tiefen Einblick in die Theorien, konzeptionellen Grundlagen, Design-Ansätze und die praktische Umsetzung der “Blended Interaction“. Diese neue Form der Interaktion mit interaktiven System ist ganz anders als traditionelle WIMP (= Windows Icons Menüs Pointer) Ansätze, wie die heutige Bedienung von grafischen Benutzeroberflächen oder Web-Seiten (mit Maus und Tastatur). Stattdessen werden neue Post-WIMP Interaktionsmodalitäten eingesetzt (z.B. Multi-Touch, Tangible User Interfaces, digitale Stifte, Gesten, Sprache), um Objekte aus der virtuellen Welt mit unserem gewohnten physischen und sozialen Arbeitsumfeld zu vermischen (unter Zuhilfenahme des Ubiquitous Computing). Dies geht weit über unsere gewohnten “Desktops“ hinaus. Allerdings wirft diese aufregende neue Welt “natürlicher“, “be-greifbarer“ und sozialer Benutzeroberflächen viele Fragen für die Wissenschaftler, Designer und Entwickler auf: Wie können stationäre Displays, physische Objekte und mobile Geräte kombiniert werden, um “vermischte“ Arbeitsumgebungen zu schaffen? Der Kurs besteht aus einem theoretischen und einem praktischen Teil. Beide werden parallel während des Semesters durchgeführt:

- Der theoretische Teil besteht aus 4 Stunden Vorlesung und Übungen pro Woche. Hier werden die Theorien, konzeptionellen Grundlagen, Modelle, Prinzipien und Hardware & Software-Technologien für die Blended Interaction erläutert.
- Der praktische Teil wird in Teams (je zwei Teilnehmer) durchgeführt. Jedes Team wird einem Projekt zugeordnet, in dem das Wissen aus dem theoretischen Teil bei der Gestaltung und Umsetzung eines Prototyps angewandt werden soll. Im Rahmen einer zweistündigen Sitzung präsentieren die einzelnen Teams ihre wöchentlichen Fortschritte vor den Teilnehmern und Betreuern des Kurses. Dies ist gleichzeitig auch die Chance Feedback und Empfehlungen von den Lehrenden und Experten der HCI Gruppe zu bekommen. Am Ende des Kurses muss das System die angegebene Aufgabe des Projekts erfüllen und darüber hinaus als interaktiver Prototyp benutzbar sein (vorzugsweise in C#/WPF; es dürfen aber auch andere Programmiersprachen eingesetzt werden)

Lernziele :

- Wie können interaktive Oberflächen oder große Displays eingesetzt werden, um eine effiziente Zusammenarbeit zwischen mehreren Benutzern zu fördern?
- Wie können die neuen Modalitäten (z.B. Touch-Gesten, Stift oder Spracheingabe) kombiniert werden, um die “natürliche“ Interaktion zu fördern? Und was bedeutet “natürliche“ oder “intuitive“ Interaktion eigentlich?



- Welche Art von Programmiermodellen und -sprachen sind nützlich, um gleichzeitige Benutzeraktivitäten zu unterstützen (z. B. gleichzeitige Eingabe mit mehreren Fingern, Gesten, physische Objekte, oder Sprache)?
- Welche Theorien und Konzepte aus der Kognitionswissenschaft, der Psychologie und den Sozialwissenschaften können uns helfen, zu verstehen und vorherzusagen, welche Auswirkungen die Blended Interaction auf die Nutzer hat?

Voraussetzung :

Zur Teilnahme an dem Kurs wird die vorherige Teilnahme an dem Kurs “Interaktive Systeme“ oder gleichwertige Erfahrung aus anderen Kursen oder Universitäten empfohlen. Bisherige Erfahrungen in der Programmierung grafischer Benutzeroberflächen oder Eingabetechnologien in C#/WPF oder Java ist ein Muss.

Arbeitsaufwand :

Insgesamt 270 Stunden, davon

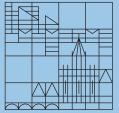
- 84 Stunden Präsenzzeit
- 186 Stunden Selbststudium

Prüfungsleistung :

- Häufige Präsenz und aktive Beteiligung in theoretischen Vorlesungen (VL), Übungen (Ü) und Projekt-Präsentationen (P). Präsenz bei allen Team-Präsentationen und der abschließenden Präsentation ist zwingend erforderlich!
- Eine schriftliche Prüfung (60 min, Multiple Choice oder ähnliches)
- Zwei Team-Präsentationen über den Verlauf des Semesters sollen die Zwischenergebnisse der praktischen Projektarbeit aufzeigen. Aktive Teilnahme an den anschließenden Diskussionen von Konzepten und deren Umsetzung.
- Finale Team-Präsentation des Ergebnisses der praktischen Projektarbeit inklusive Diskussion der Konzepte und Umsetzung derer.
- Abgabe einer selbsterklärenden Präsentation. Hierbei sollen der Entwurf und die Umsetzung des Projekts beschrieben sein und es sollte auf die aufgetretenen Probleme und deren Lösung eingegangen werden. Insbesondere muss die Präsentation Bezug zu verwandten Arbeiten, wie wissenschaftlichen Zeitschriften, Büchern und Konferenzbeiträgen und verwandten Designs von kommerziellen Produkten, aufzeigen.
- Die Gesamtnote wird aus den Noten für die Qualität der Vorträge (20%), die Qualität des interaktiven Prototypen (40%), und der individuellen Note der schriftlichen Prüfung (40%) berechnet.

Literatur :

- Imaz, M., Benyon D. (2007). *Designing with Blends - Conceptual Foundations of Human-Computer Interaction and Software Engineering*. MIT Press. [recommended are chapters 1 to 4]



15.6 Factorization Models for Machine Learning

Modulverantwortlicher : Jun.-Prof. Dr. Rendle (z.B. INF-12470-20131)

Credits : 4 SWS, 6 ECTS

Inhalte :

- Modelle: Matrixfaktorisierung, höherstufige Faktorisierungsmodelle, hierarchische Modelle
- Lernmethoden und Regularisierungsansätze mit Gradientenverfahren und Bayesscher Inferenz
- Anwendungen: Empfehlungssysteme, Soziale Netzwerkanalyse

Lernziele :

Die Studierenden verstehen die bekanntesten Faktorisierungsansätze und deren Lernmethoden. Die Studierende können Faktorisierungsmodelle einordnen und beurteilen, welche Methoden für bestimmte Problemstellungen geeignet sind. Sie verstehen, wie die Methoden in einem Computer umgesetzt werden können. Die Teilnehmer sind in der Lage die vorgestellten Konzepte wie Maximum-Likelihood-Schätzer auf andere Modelle zu transferieren und dazu Lernalgorithmen zu entwickeln.

Voraussetzung :

Die Vorlesung ist für Teilnehmer/innen im Vertiefungs- und Masterstudium offen. Grundkenntnisse in Data Mining oder Machine Learning sind von Vorteil.

Arbeitsaufwand :

180 Stunden, davon ca. 60 Stunden Präsenzstudium und 120 Stunden Selbststudium

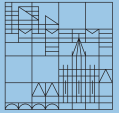
Prüfungsleistung :

mündliche Prüfung

Literatur :

Wird pro Kapitel in der Vorlesung vorgestellt. Folgende zwei Bücher sind als allgemeine Nachschlagewerke zu maschinellem Lernen und Bayesscher Datenanalyse empfehlenswert:

- Gelman et al.: *Bayesian Data Analysis*, 2nd Edition
- Hastie et al.: *The Elements of Statistical Learning*, 2nd Edition



15.7 Datenkompression (Data Compression)

Modulverantwortlicher : Prof. Dr. Saupe (z.B. INF-11520-20101)

Credits : 6 SWS, 9 ECTS

Inhalte :

Die Vorlesung bietet eine Einführung in die theoretischen Grundlagen der Datenkompression und die wichtigsten Verfahren der Text-, Bild- und Videokompression.

- Quellcodierung: die Theoreme von McMillan und Kraft
- Entropie und die Fundamentalsätze von Shannon
- Huffman-Codierung
- Arithmetische Codierung, Kontext-Modellierung
- Wörterbuch-Methoden
- Lauflängen-Codierung
- Prädiktionscodierung
- Skalare- und Vektorquantisierung
- Diskrete-Cosinus-Transformation und JPEG
- Wavelet-Codierung (EZW, SPIHT, EBCOT, JPEG2000)
- Video-Codierung (MPEG, H.263)

Lernziele :

Die Studierenden haben ein Grundverständnis der Möglichkeiten und Grenzen der Datenkompression von Text, Bildern und Bildsequenzen, eine Übersicht über die wichtigsten Verfahren und Kenntnis ihrer Arbeitsweise sowie die Fähigkeit zum Einsatz von Standardverfahren.

Voraussetzung :

Es werden Vorkenntnisse in elementarer Wahrscheinlichkeitstheorie, linearer Algebra sowie Algorithmen und Datenstrukturen erwartet. Programmierkenntnisse sind erforderlich.

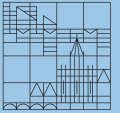
Arbeitsaufwand :

- ECTS: 9 Credits
- SWS: 3 Stunden Vorlesung, 3 Stunden Übung
- Insgesamt 270 Stunden, davon 84 Stunden Präsenzzeit

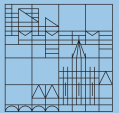
Prüfungsleistung :

Übungsaufgaben, Klausur

Literatur :



- Sayood, K., *Introduction to Data Compression*, Morgan Kaufmann Publishers, 1996.
- Taubman, D. S., Marcellin, M., *Jpeg2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, 2002.
- Witten, I. H., Moffat, A., Bell, T. C., *Managing Gigabytes*, Morgan Kaufmann Publishers, 1999.



15.8 Mustererkennung (Pattern Recognition)

Modulverantwortlicher : Prof. Dr. Saupe (z.B. INF-10920-20111)

Credits : 6 SWS, 9 ECTS

Inhalte :

The aim of this course is to provide students with a first introduction to the statistical inference theory that is fundamental to the area of pattern recognition. The main topics include Bayes classification, construction of linear classifiers, parameter estimation and nonparametric estimation of probability density functions, classifier evaluation and feature selection. The lectures are accompanied with practical lab sessions, giving students an opportunity to apply the acquired knowledge doing both paper-and-pencil and computer-based (e.g., in MATLAB) exercises.

Lernziele :

The aim of this one-semester course is to provide students with a first introduction to the statistical inference theory that is fundamental to the area of pattern recognition. The main topics include Bayes classification, construction of linear classifiers, parameter estimation and nonparametric estimation of probability density functions, classifiers evaluation and feature selection. The lectures are accompanied with practical lab sessions, giving students an opportunity to apply the acquired knowledge doing both paper-and-pencil and computer-based (e.g., in MATLAB) exercises.

Voraussetzung :

Grundkenntnisse in Stochastik, Analysis und Lineare Algebra.

Arbeitsaufwand :

- ECTS: 9 Punkte.
- SWS: 3 Stunden Vorlesung, 3 Stunden Übung.
- Insgesamt: 270 Stunden, davon 84 Stunden Präsenzstudium.

Prüfungsleistung :

Übungsaufgaben, Klausur

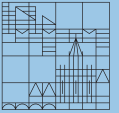
Literatur :

Hauptreferenz:

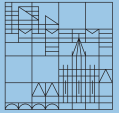
- Richard O. Duda, Peter E. Hart and David G. Stork. *Pattern Classification* (2nd ed.), 2000.

Weitere Literatur:

- Christopher M. Bishop. *Pattern Recognition and Maschine Learning*, 2007.
- Keinosuke Fukunaga. *Introduction to Statistical Pattern Recognition* (2nd edition), 1990.
- Sergios Theodoridis, Konstantinos Koutroumbas. *Pattern Recognition* (3d edition), 2006.



- Lawrence R. Rabiner (February 1989). *A tutorial on Hidden Markov Models and selected applications in speech recognition*. Proceedings of the IEEE 77 (2): 257-286.



15.9 Digital Libraries (Digital Libraries)

Modulverantwortlicher : Jun.-Prof. Dr. Schreck (z.B. INF-10005-20122)

Credits : 3 SWS, 5 ECTS

Inhalte :

- Einführung in Geschichte, Anwendungen und Herausforderungen von Digitalen Bibliotheken
- Aktuelle Digital Library Systeme
- Referenzmodelle und Architekturen
- Metadaten und Inhaltserschließung
- Textuelle und nichttextuelle Dokumente
- Einführung in Multimedia Retrieval
- Digitalisierung und Langzeitarchivierung
- Nutzersicht: Visuelle Suche, Dokumentvisualisierung, Personalisierung
- Evaluation, User Studies
- Einführung in aktuelle Forschungsthemen

Lernziele :

Die Studierenden werden in Kerntechnologien für den Aufbau und Betrieb von Digitalen Bibliothekssystemen eingeführt. Es werden Kenntnisse über Methoden zur automatischen Inhaltserschließung, Suche und Präsentation von ausgewählten textuellen bzw. nichttextuellen Dokumenttypen vermittelt. Die Studierenden erlangen ein Verständnis für die Problemfelder der Langzeitarchivierung von digitalen Dokumenten und entsprechende Lösungsmöglichkeiten kennen. Die Studierenden werden in die Lage versetzt, Anforderungen an digitale Bibliothekssysteme aus Nutzersicht zu beurteilen und entsprechende Systeme zu konzipieren.

Voraussetzung :

Grundvorlesungen des Bachelorstudienganges.

Arbeitsaufwand :

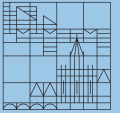
Insgesamt 150 Stunden, verteilt auf 42 Stunden Präsenzstudium und 108 Stunden Eigenstudium.

Prüfungsleistung :

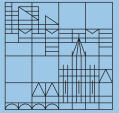
Studienleistung: 50% der Punkte aus den zweiwöchentlichen Übungen. Prüfung: je nach Anzahl Teilnehmer mündliche (Einzelprüfung von 30 Minuten Dauer) oder schriftliche (Klausur von 120 Minuten Dauer) Prüfung. Die Note ergibt sich aus der Note der Prüfung.

Literatur :

Die Vorlesung basiert auf einem Folienskript, das zur Verfügung gestellt wird. Die Inhalte basieren auf einer Reihe von Artikeln und Lehrbüchern. Zur Vorbereitung wird z.B. die nachfolgende Literatur empfohlen:



- Michael Lesk: *Understanding Digital Libraries*, Morgan Kaufmann, 2005.
- Ian H. Witten, David Bainbridge, David M. Nichols: *How to Build a Digital Library*, Morgan Kaufmann, 2009.
- J. Zhang: *Visualization for Information Retrieval*, Springer, 2008.
- *Bestandsaufnahme zur Digitalisierung von Kulturgut und Handlungsfelder*. Studie des Fraunhofer Instituts für Intelligente Analyse- und Informationssysteme, 2007.
- B. Haslhofer and W. Klas: *A Survey of Techniques for Achieving Metadata Interoperability*, ACM Computing Surveys, 2010.



15.10 MindSwarms 2.0 - Swarm Networks

Modulverantwortlicher : Prof. Dr. Waldvogel (z.B. INF-12100-20111)

Credits : 4 SWS, 6 ECTS

Inhalte :

In dieser Lehrveranstaltung werden Roboterschwärme und deren dynamische Netzwerkstrukturen untersucht. Es werden zunächst grundlegende Aspekte besprochen und praktisch umgesetzt, die gleichzeitig die Andersartigkeit der Entwicklung von Programmen zur Steuerung von Robotern hervorheben. So werden zum Beispiel verschiedene Aspekte aus der Automatentheorie besprochen (Finite State Machines – FSM) und umgesetzt. Der Schwerpunkt der Vorlesung liegt jedoch auf Konzepten selbstorganisierender, mobiler Netzwerke und der praktischen Umsetzung dieser Netzwerke. Hervorzuheben ist, dass die praktische Umsetzung im Rahmen eines Projekts stattfindet, bei der zunächst Teams von je drei bis fünf Studenten kleinere Problemstellungen lösen. Im Laufe der Veranstaltung werden diese kleineren Teams dann immer stärker miteinander interagieren, um am Ende gemeinsam an der finalen Aufgabe zu arbeiten.

Voraussetzung :

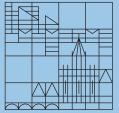
Als Voraussetzung sollten Studierende (sehr) gute Programmierkenntnisse in Java mitbringen sowie die Motivation, sich von Grund auf in eine neue Programmierbibliothek (LeJos) einzuarbeiten. Weiter sollten alle Teilnehmer eine gesunde Portion Experimentierfreude mitbringen, da sich die Entwicklung von Programmen zur Steuerung von Robotern sowie deren Kollaboration deutlich von der normalen Softwareentwicklung abhebt (z.B. das Testen von Sensoren und deren Möglichkeiten). Da die Lehrveranstaltung unter anderem eine Team-Projektarbeit umfasst, sollte das Softwareprojekt bereits besucht worden sein. Das Engagement sich mit Problemen und Aufgabenstellungen auseinanderzusetzen, die über die aus der Softwareentwicklung bekannten Hürden hinausgehen, wird als selbstverständlich vorausgesetzt. Der Besuch der Vorlesung “MindSwarms:: Swarm Communication“ aus dem letzten Jahr ist keine Voraussetzung für diese Vorlesung, kann sich jedoch als hilfreich erweisen.

Arbeitsaufwand :

Präsenzstudium: 56 Stunden, Eigenstudium: 124 Stunden

Prüfungsleistung :

Der Leistungsnachweis besteht aus einem Projektbericht, der jeweils in Teams zu drei bis fünf Studenten gemeinsam verfasst wird, sowie der finalen Vorstellung des Projekts in einer abschließenden Projektbesprechung sowie dem Quellcode (inkl. Quellcode-Dokumentation) des Teams. Der Projektbericht wird sukzessiv während des Semesters geschrieben und umfasst unter anderem die Lösungen zu gestellten Teilaufgaben sowie Beschreibungen/Berichte zu Ideen und Experimenten. Der Projektbericht eines Teams soll ca. 24 bis 40 Seiten lang sein (je nach Größe des Teams, also ca. 8 Seiten pro Person).



15.11 Data Mining: Foundations (Data Mining: Foundations)

Modulverantwortlicher : Prof. Dr. Berthold (z.B. INF-11430-20122)

Credits : 4 SWS, 8 ECTS

Inhalte :

The lecture series provides an introduction to Data Mining Methods with an emphasis placed on basic approaches and how they are incorporated into different problem definitions.

- Data Mining: problem definition, motivation, application examples
- Modelling: data-driven concept development, presentation of hypotheses
- Version space and the evaluation of hypotheses
- Clustering methods
- Regression
- Association rules

Lernziele :

Achieve an understanding of data mining. This includes the theoretical concepts which are for example the data mining cycle, but also the algorithms to explore the data.

Voraussetzung :

Basic mathematical skills (statistics), basic knowledge in programming (JAVA), basic courses in computer science.

Arbeitsaufwand :

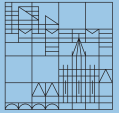
Präsenzstudium: 56 Stunden, Eigenstudium: 184 Stunden

Prüfungsleistung :

There will be programming exercises and theoretical exercises which are a requirement for taking the exam. The final grade will only be based on the oral exam (20 minutes) at the end of the semester.

Literatur :

- Berthold, M.R., Borgelt, C., Höppner, F., Klawonn, F., *Guide to intelligent data analysis: How to intelligently make sense of real data*, Springer, 2010
- Han J., Kamber M., *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, August 2000.
- Ester M., Sander J., *Knowledge Discovery in Databases. Techniken und Anwendungen*, Springer, 2000.
- Hand D.J., Mannila H., Smyth P., *Principles of Data Mining*, MIT Press, 2001.
- Mitchell T. M., *Machine Learning*, McGraw-Hill, 1997.
- Witten I. H., Frank E., *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann Publishers, 2000.



15.12 Data Mining: Artificial Intelligence (Data Mining: Artificial Intelligence Methods)

Modulverantwortlicher : Prof. Dr. Berthold (z.B. INF-10320-20131)

Credits : 2 SWS, 4 ECTS

Inhalte :

This lecture covers special data mining methods in the area of soft computing and artificial intelligence.

- Introduction to Fuzzy Logic
- Learning from Fuzzy Logic Models
- Introduction to Neural Networks
- Learning and Analysis of Neural Networks
- Introduction to Metaheuristics
- Evolution of Rule- and other Models
- Hybrid Methods of Soft Computing in Data Mining

Lernziele :

The student should gain a deeper understanding how data can be modeled in the area of artificial intelligence. Here the Fuzzy modeling mainly yields in a better understanding of data. Metaheuristic give an easy approach to solve problems which did not seem solvable before. Finally the neural networks provide a very powerful tool to model complex sets of data.

Voraussetzung :

Basic knowledge within the content of the *Data Mining 1* lecture respectively the book *Guide to Intelligent Data Analysis* is advantageous.

Arbeitsaufwand :

Präsenzstudium: 28 Stunden, Eigenstudium: 92 Stunden

Prüfungsleistung :

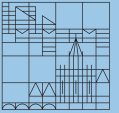
Active participation in the exercise: vote for at least 60% of the exercise, that is to agree to show the solution approach on the black board. Oral (30 minutes) exam. The final mark is the mark of the exam.

Literatur :

For all topics:

- Michael Berthold, David Hand: *Intelligent Data Analysis, An Introduction*, 2te Auflage, Springer-Verlag, 2003.
- Online book *Fuzzy Logic*, (german) partly very theoretical

Book about neuronal networks (and fuzzy systems) (german):



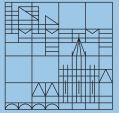
- Detlef Nauck, Christian Borgelt, Frank Klawonn und Rudolf Kruse.: *Neuro-Fuzzy-Systeme - Von den Grundlagen Neuronaler Netze zu modernen Fuzzy-Systemen*, Vieweg-Verlag, Wiesbaden, Germany 2003, ISBN 3-528-25265-0

Very detailed slide based script:

- Christian Borgelt *Introduction to Neural Networks*, <http://www.borgelt.net/slides/nn.pdf>

Another script to neuronal networks (german):

- *Ein kleiner Überblick über Neuronale Netzwerke*



15.13 Netzwerkanalyse (Network Analysis)

Modulverantwortlicher : Prof. Dr. Brandes (z.B. INF-2405-20121)

Credits : 6 SWS, 9 ECTS

Inhalte :

Netzwerke sind heutzutage allgegenwärtig. Neben physisch realisierten Netzwerken wie z.B. in der Elektrotechnik oder dem Transportwesen werden zunehmend auch abstrakte Netzwerke wie z.B. die Verbindungsstruktur des WWW oder Konstellationen politischer Akteure analysiert. Bedingt durch die Vielzahl der Anwendungen und resultierenden Fragestellungen kommt dabei ein reicher Methodenkatalog zur Anwendung, der auf interessante Zusammenhänge zwischen Graphentheorie, Linearer Algebra und probabilistischen Methoden führt. In dieser Veranstaltung sollen einige der eingesetzten Methoden und deren Grundlagen systematisch behandelt werden. Fragestellungen werden zwar exemplarisch an Anwendungsbeispielen motiviert, der Schwerpunkt wird aber auf den zur Lösung verwendeten algorithmischen Vorgehensweisen sowie deren Voraussetzungen und Eigenschaften liegen. Darüber hinaus werden wir auf zahlreiche offene Fragen stoßen, die zu einer vertieften Beschäftigung mit dem Thema einladen.

Voraussetzung :

Grundlegende Kenntnisse in Algorithmik und diskreter Mathematik (etwa im Umfang der Pflichtveranstaltungen "Algorithmen und Datenstrukturen" sowie "Diskrete Mathematik" im Bachelorstudium).

Arbeitsaufwand :

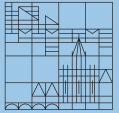
270 Stunden, davon ca. 90 Stunden Präsenzstudium und 180 Stunden Selbststudium

Prüfungsleistung :

Je nach Teilnehmer(innen)zahl mündliche Prüfung von 30min oder Klausur von 120min. Erfolgreiche Teilnahme an den Übungen (Studienleistung) ist Voraussetzung für das Ablegen der Prüfung.

Literatur :

- Brandes, Erlebach (Hrsg.): *Network Analysis*. LNCS 3418, Springer, 2005.
- Diestel: *Graphentheorie*. Springer, 1996.
- Kolaczyk: *Statistical Analysis of Network Data*. Springer, 2009.
- Steger: *Diskrete Strukturen*. Springer, 2001.
- Wasserman, Faust: *Social Network Analysis*. Cambridge Univ. Press, 1994.



15.14 Zeichnen von Graphen (Graph Drawing)

Modulverantwortlicher : Prof. Dr. Brandes (z.B. INF-2295-20111)

Credits : 6 SWS, 9 ECTS

Inhalte :

Grundlage für die effektive und effiziente Visualisierung von Netzwerken sind Algorithmen zur Bestimmung eines Layouts für den die Netzwerkstruktur beschreibenden Graphen. Das automatische Zeichnen von Graphen hat daher wichtige Anwendungen in Kernbereichen der Informatik wie etwa Datenbanken, Software Engineering, VLSI- und Netzwerk-Design und visuelle Benutzerschnittstellen. Anwendungen in anderen Bereichen betreffen alle Aspekte der visuellen Datenanalyse, z.B. in den Ingenieurwissenschaften, Chemie und Biologie oder Sozial- und Politikwissenschaft. Dazu werden verschiedene algorithmische Prinzipien und Methoden wie z.B. kräftebasierte Verfahren und Flussmethoden besprochen.

Arbeitsaufwand :

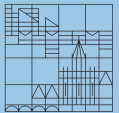
Präsenzstudium: 84 Stunden, Eigenstudium: 186 Stunden

Prüfungsleistung :

mündliche Prüfung

Literatur :

- G. Di Battista, P. Eades, I.G. Tollis, R. Tamassia: *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, 1999.
- M. Jünger, P. Mutzel (Eds.): *Graph Drawing Software*. Mathematics and Visualization Series, Springer-Verlag, 2003.
- M. Kaufmann, D. Wagner (Eds.): *Drawing Graphs – Methods and Models*. Lecture Notes in Computer Science Tutorial 2025, Springer-Verlag, 2001.
- K. Sugiyama: *Graph Drawing and Applications for Software and Knowledge Engineers*. World Scientific, 2002.



15.15 Entwurf und Analyse von Algorithmen (Design and Analysis of Algorithms)

Modulverantwortlicher : Prof. Dr. Brandes (z.B. INF-10160-20092)

Credits : 6 SWS, 9 ECTS

Inhalte :

Entwurf und Analyse von Algorithmen sind zentrale Aufgaben der Informatik. Diese Vorlesung behandelt wichtige Fragestellungen und Methoden der Algorithmik und schafft eine allgemeine Grundlage für die Beschäftigung mit spezielleren algorithmischen Problemen. Es werden Algorithmen und Datenstrukturen aus verschiedenen Bereichen und insbesondere Graphenalgorithmen behandelt.

Voraussetzung :

Kenntnis etwa im Umfang von "Algorithmen und Datenstrukturen"

Arbeitsaufwand :

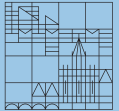
Präsenzstudium: 84 Stunden, Eigenstudium: 186 Stunden

Prüfungsleistung :

Teilnahme an den Übungen (50% der Punkte) und mündliche Prüfung, 1. Termin: Mi, 10.02. 2010 ab 8.00 (E 212), 2. Termin: Mo, 12.04. 2010 ab 14.00 (E 212)

Literatur :

- T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein: *Introduction to Algorithms*. McGraw-Hill, 2001 (2nd ed.)
- J. Kleinberg, E. Tardos: *Algorithm Design*. Addison-Wesley, 2006
- M.T. Goodrich, R. Tamassia: *Algorithm Design*. Wiley, 2002
- T. Ottmann, P. Widmayer: *Algorithmen und Datenstrukturen*. BI-Wissenschaftsverlag, 1993
- D. Jungnickel: *Graphen, Netzwerke und Algorithmen*. BI-Wissenschaftsverlag, 1994



15.16 Modellierung in der Computergrafik (Modelling in Computer Graphics)

Modulverantwortlicher : Prof. Dr. Deussen (z.B. INF-12740-20131)

Credits : 4 SWS, 6 ECTS

Inhalte :

mit Methoden der Datenmodellierung in der Computergrafik. Ausgehend von Volumen- und Oberflächenmodellen, deren Herstellung, Eigenschaften und Bilderzeugung werden insbesondere komplexe und datengetriebene Modelliermethoden besprochen.

- Geometrische Repräsentationsformen
- Kurven und Flächen, Parametrisierungsverfahren
- Fraktale
- Botanik und computergenerierte Pflanzen
- Modellierung virtueller Ökosysteme
- Datengetriebene Modellgenerierung (Rekonstruktion aus Punktdaten)
- Level-of-Detail Methoden

Lernziele :

Absolventen des Kurses haben vertiefte Kenntnisse zu computergrafischen Modellierverfahren und sind in der Lage, komplexe Szenen aufzubauen und zu verwalten. Sie haben vertieftes Wissen über Rastergrafikmethoden und können den Erstellungsaufwand für komplexe Computergrafiken abschätzen.

Voraussetzung :

Einführungsveranstaltung Computergrafik, Computergrafik und interaktive Systeme, Programmierkenntnisse

Arbeitsaufwand :

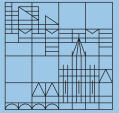
90 Stunden (26 Stunden Präsenzzeit, 64 Stunden Eigenstudium)

Prüfungsleistung :

mündliche Prüfung 20min oder 90min Klausur (wird in der Vorlesung bekannt gegeben)

Literatur :

wird im Kurs bekannt gegeben



15.17 Methoden des Echtzeitrendering (Methods for Realtime Rendering)

Modulverantwortlicher : Prof. Dr. Deussen (z.B. INF-12280-20112)

Credits : 4 SWS, 6 ECTS

Inhalte :

Hardwarenahe Verfahren der Computergrafik: insb. Echtzeitrendering, Schattenberechnung, Texturing (Multitexturing, Lightmaps), Image-Based Rendering, hardwarenaher Level-of-detail, Shader-programming. Die Vorlesung ist eine Spezialveranstaltung für Studenten, die die einführenden Grafikveranstaltungen schon gehört haben. Das zugrundeliegende Buch wird gemeinsam durchgearbeitet, die Studierenden müssen in Kurzreferaten die Kapitel zusammenfassen.

Lernziele :

Absolventen des Kurses sind in der Lage, komplexe hardware-nahe Problemstellungen der Computergrafik eigenständig zu analysieren, zu bewerten und auch zu implementieren. Solche graphischen Systeme werden in der Spieleprogrammierung verwendet, im Bereich Simulation und Visualisierung, an allen Stellen, wo Echtzeitgrafik erforderlich ist.

Voraussetzung :

Bachelor Information Engineering, Informatik oder ähnlich. Grundvorlesung Computergrafik oder Computergrafik und Interaktive Systeme sowie Vorlesung "3D Computergrafik und Objekmodellierung"

Arbeitsaufwand :

Insgesamt 180 Stunden

- 54 Stunden Präsenzstudium
- 124 Stunden Eigenstudium

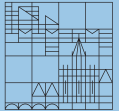
Prüfungsleistung :

Prüfung: Je nach Teilnehmer/innen/zahl mündlich (Einzelprüfung von 20 Minuten Dauer) oder schriftlich (Klausur von 60 Minuten Dauer).

Note: Die Note ergibt sich aus der Note der Prüfung.

Literatur :

- Buch "Real-Time rendering" von Thomas Akenine-Moller, Eric Haines, Naty Hoffman. ISBN 978-1568814247



15.18 Illustrative Computergrafik (Illustrative Computer Graphics)

Modulverantwortlicher : Prof. Dr. Deussen (z.B. INF-11540-20112)

Credits : 4 SWS, 6 ECTS

Inhalte :

- 2D-Methoden zur Erzeugung nicht-fotorealistischer Computergrafiken (Dithering, Halftoning, artistic screening, Stipping)
- Bearbeitung von 2.5dimensionalen Daten (Bildern mit Tiefeninformationen), Kantenverstärkung, Tiefendifferenzen, Unsharp Masking
- Nicht-fotorealistische Darstellung von 3D Geometriedaten (Bestimmung geometrischer Features, Minima und Maxima von Ableitungen und Krümmungen, Silhouettenfindung, numerische Probleme)

Lernziele :

Verstehen, Analysieren und Bewerten von Methoden der Nicht-Fotorealistischen Computergrafik. Die Vorlesung behandelt 2D-, 2.5D- und 3D-Methoden zur Erzeugung von Computerbildern jenseits der fotorealistischen Bilderzeugung. Absolventen des Kurses sind in der Lage, künstlerische Abstraktionsmethoden zu implementieren, zu evaluieren und zu analysieren. Im Kurs wird die Implementierung in Java (innerhalb des Processing Frameworks) praktisch durchgeführt.

Voraussetzung :

Entsprechend den Modulen Mathematische Grundlagen des Information Engineering und Methoden der Praktischen Informatik, elementare Programmierkenntnisse. Besuch der Vorlesung Computergrafik und interaktive Systeme oder andere Computergrafik-Grundlagenvorlesung

Arbeitsaufwand :

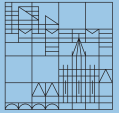
Präsenzstudium: 56 Stunden, Eigenstudium: 124 Stunden

Prüfungsleistung :

Studienleistung: 50% der Punkte aus den Übungen
Prüfung: Je nach Teilnehmer/innen/zahl mündlich (Einzelprüfung von 20 Minuten Dauer) oder schriftlich (Klausur von 60 Minuten Dauer)
Note: Die Note ergibt sich aus der Note der Prüfung.

Literatur :

wird im Kurs ausgegeben



15.19 3D-Computergraphik und Objektmodellierung (3D Computer Graphics and Object Modelling)

Modulverantwortlicher : Prof. Dr. Deussen (z.B. INF-2048-20112)

Credits : 4 SWS, 6 ECTS

Inhalte :

- Raytracing (Basic Algorithm, Basic Shapes and Ray/Primitive Intersection, Data Structures for RT (BVH, BSP-Trees, Octrees, LOD))
- Light Transport (Light Transport and Global Illumination, Sampling, Monte-Carlo Integration, Reflection Models and Materials, Path Tracing, Particle Tracing, Radiosity)
- Modeling (Meshes, Implizit, CSG, Rule based, physical Modeling)
- Discretisation (FEM, Points, Particle)
- Level-of-Detail modeling
- Image-Based Rendering, Texture Synthesis (Graphcut, Tile-based, By-example)

Lernziele :

Verstehen, Analysieren und Bewerten von fortgeschrittenen Algorithmen der Computergrafik. Hierbei insbesondere effiziente Verfahren der Geometriemodellierung, Level-of-detail Methoden, globale Beleuchtungsverfahren und Texturierung, die es ermöglichen, komplexe Objekte zu bearbeiten und darzustellen. Die Absolventen werden in der Lage sein, komplexe Computergrafiksysteme zu realisieren und zu evaluieren.

Voraussetzung :

Entsprechend den Modulen Mathematische Grundlagen des Information Engineering und Methoden der Praktischen Informatik, elementare Programmierkenntnisse. Besuch der Vorlesung "Computergraphik und interaktive Systeme" oder andere Computergrafik-Grundlagenvorlesung

Arbeitsaufwand :

Insgesamt 180 Stunden, verteilt auf:

- 56 Stunden Präsenzstunden
- 124 Stunden Eigenstudium

Prüfungsleistung :

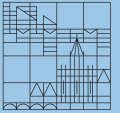
Studienleistung: 50% der Punkte aus den Übungen.

Prüfung: Je nach Teilnehmer/innen/zahl mündlich (Einzelprüfung von 20 Minuten Dauer) oder schriftlich (Klausur von 60 Minuten Dauer).

Note: Die Note ergibt sich aus der Note der Prüfung.

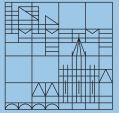
Literatur :

- P. Shirley: *Fundamentals of Computer Graphics*, A K Peters, Ltd. ISBN: 9781568812694



- A. Watt, M. Watt: *Advanced Animation and Rendering Techniques*, Addison Wesley, ISBN 0-201-54412-1
- O. Deussen: *Computergenerierte Pflanzen*, Springer-Verlag, ISBN 3-540-43606-5

Weitere Literatur wird im Kurs bekannt gegeben.



15.20 Multimedia database systems (Multimedia database systems)

Modulverantwortlicher : Prof. Dr. Keim (z.B. INF-2287-20121)

Credits : 5 SWS, 8 ECTS

Inhalte :

- Grundlagen von Multimedia-Datenbanksystemen
- Ähnlichkeitsmodelle für Bilder
- Geometrische Ähnlichkeitsmodelle
- 3D-Model Retrieval
- Exkurse zu weiteren Multimedia-Datenarten und Spezialanwendungen, z.B. Audiodaten, Zeitseriendaten und Biometrie (z.B. Gesichtserkennung, Fingerabdruckerkennung).
- Algorithmen für Bereichs- und kNN-Anfragen
- Indexstrukturen für hochdimensionale Vektorräume
- Metrische Indexstrukturen

Lernziele :

Die Studierenden werden in die Lage versetzt, inhaltsbasierte Zugriffsmethoden für eine Reihe unterschiedlicher Multimedia-Datentypen wie z.B. Bilder, Video, 3D-Objekten oder Zeitseriendaten zu verstehen und hinsichtlich ihres Einsatzes in Multimedia-Datenbanksystemen beurteilen zu können. Sie werden in die Lage versetzt, Algorithmen zur Bearbeitung von Ähnlichkeitssuchanfragen in Multimedia-Datenbanken zu konzipieren und weiterzuentwickeln. Darüber hinaus werden Kenntnisse über spezialisierte Indexstrukturen zur effizienten Anfragebearbeitung behandelt. Die Teilnehmer werden dadurch in die Lage versetzt, Effizienzanforderungen zu beurteilen und vorhandene Verfahren adaptieren und anwenden zu können.

Voraussetzung :

Methoden der Praktischen Informatik, Informationsmanagement, Informationssysteme

Arbeitsaufwand :

Insgesamt 240 Stunden, verteilt auf

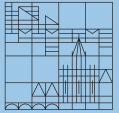
- 70 Stunden Präsenzstudium
- 170 Stunden Eigenstudium

Prüfungsleistung :

Studienleistung: Aktive Teilnahme an den Übungen

Prüfung: je nach Teilnehmer/innen/zahl mündlich (Einzelprüfung von 30 Minuten Dauer) oder schriftlich (Klausur von 120 Minuten Dauer). Für die Zulassung zur Prüfung müssen 50% der Punkte in den Übungen erreicht werden.

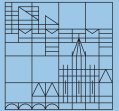
Note: Die Note ergibt sich aus der Prüfungsnote.



Literatur :

Die Vorlesung basiert auf einem Skript und einer Reihe von englischsprachigen Konferenz- und Zeitschriftenartikeln, die zu den jeweiligen Themengebieten behandelt werden. Vorbereitend kann folgende Literatur empfohlen werden:

- Rüger, *Multimedia Information Retrieval*. Morgan and Claypool Publishers 2010.
- I. Schmitt, *Ähnlichkeitssuche in Multimedia-Datenbanken*. Oldenbourg, 2006.
- Blanken, de Vries, Blok, Feng, *Multimedia Retrieval*. Springer 2007
- Volker Gaede, Oliver Günther, *Multidimensional Access Methods*. In: ACM Computing Surveys, 30 (2): 170-231 (1998)
- Christian Böhm, Stefan Berchtold, Daniel A. Keim, *Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases*. In: ACM Computing Surveys, 33(3): 322-373 (2001)



15.21 Logic in Computer and Software Science

Modulverantwortlicher : Prof. Dr. Leue (z.B. INF-10825-20102)

Credits : 4 SWS, 6 ECTS

Inhalte :

Formal logic is the foundation for reasoning about many computer and software systems. For instance, they are used when reasoning about the correctness of software, when specifying queries on data bases, or to represent knowledge in artificial intelligence. The objective of this course is to convey formal logic as an engineering calculus in the design of computing systems. Amongst others, this course will touch on the following topics: propositional logic, natural deduction, satisfiability checking, predicate logic, second order logic for software specification, Temporal Logic, CTL model checking, Hoare Logic, proof calculi, binary decision diagrams, symbolic model checking, and further advanced topics.

Lernziele :

The goal of this course is to enable participants to choose adequate logics and use them according to their capabilities in practical computing system design situations and when pursuing further research in this area.

Voraussetzung :

Bachelor level knowledge in algorithms, theory, and programming. This course is designed for master and PhD students, although interested bachelor or Lehramt students are welcome as well.

Arbeitsaufwand :

180 hrs (6 ECTS).

Prüfungsleistung :

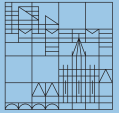
Final written examination (90 mins.) or oral examination (30 mins.)

Literatur :

Primary course text:

- M. Huth and M. Ryan, *Logic in Computer Science*, Cambridge University Press, Second Edition, 2004.

Further literature will be announced during the course.



15.22 Model Checking of Software and Systems

Modulverantwortlicher : Prof. Dr. Leue (z.B. INF-12110-20111)

Credits : 6 SWS, 10 ECTS

Inhalte :

The course will introduce into model checking for reactive software and systems. Model checking is an algorithmic, automated technique for the behavioral analysis of soft- and hardware systems. We will illustrate the algorithmic foundations of this technique, present the SPIN tool, and address advanced topics. The accompanying project will aim at the modeling and analysis of an industrial case-study.

Lernziele :

It is the goal of the course to enable students to understand and assess the efficiency and practical applicability of model checking technology. It is a further goal to illustrate to students that model checking is implemented in tools, and that these can be applied to practical, industrial-strength case studies. This will enable them to decide about an adequate use of model checking technology when conducting their own software and systems development projects, or to pursue further research in this area.

Voraussetzung :

Bachelor level background in programming, algorithms, concurrent systems, automata theory, graph theory, and logic.

Arbeitsaufwand :

300 hrs (10 ECTS).

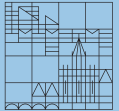
Prüfungsleistung :

Final written examination (90 mins) or oral examination (30 mins.) Participants achieving at least 80% of the marks that can be obtained in the assignments will earn a 0.3 point grade bonus.

Literatur :

- E. Clarke, O. Grumberg and D. Peled, *Model Checking*, MIT Press, 1999
- C. Baier and J.-P. Katoen, *Principles of Model Checking*, MIT Press, 2008
- G. Holzmann, *The SPIN Model Checker - Primer and Reference Manual*, Addison Wesley, 2003

Further literature will be announced during the course.



15.23 Advanced Model Checking (Advanced Model Checking)

Modulverantwortlicher : Prof. Dr. Leue (z.B. INF-12680-20131)

Credits : 5 SWS, 8 ECTS

Inhalte :

The course will cover advanced topics in Model Checking. The course will introduce into CTL model checking and into the foundations of BDD based symbolic model checking. The course will further cover model checking of real-time and probabilistic systems. The course Model Checking of Software and Systems is not a prerequisite for the successful completion of this course.

Lernziele :

It is the objective of this course to enable students to assess and use the particular model checking technologies that will be presented in this course. The participants of this course will be enabled to use model checking technologies in practical problems and to estimate and anticipate the possibilities as well as the limitations of using this technology in practical systems engineering projects. They will also be enabled to choose adequate tool support for the solution of practical model checking problems.

Voraussetzung :

A good undergraduate level knowledge of automata theory, algorithmics, and logic. The course Model Checking of Software and Systems is not a prerequisite for this course.

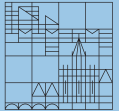
Arbeitsaufwand :

240 hrs (8 ECTS).

Literatur :

- E. Clarke, O. Grumberg and D. Peled, *Model Checking*, MIT Press, 1999
- C. Baier and J.-P. Katoen, *Principles of Model Checking*, MIT Press, 2008

Further literature will be announced during the course.



15.24 Data Warehousing und OLAP (Data Warehousing and OLAP)

Modulverantwortlicher : Prof. Dr. Scholl (z.B. INF-10365-20122)

Credits : 4 SWS, 6 ECTS

Inhalte :

Die Veranstaltung behandelt den Aufbau und Betrieb von Data Warehouse Systemen, die Abgrenzung zu Datenbanksystemen (für OLTP: Online Transaction Processing), Tools für Online Analytical Processing (OLAP) und den Extract-Transform-Load- (ETL-) Prozess, mit dem Daten aus operativen Systemen in ein Data Warehouse geladen werden. Die Vorlesung gliedert sich grob wie folgt:

- Introduction & Terminology
- Data Warehouse Architecture
- Multidimensional Data Model
- Modeling Methodologies
- Extraction, Transformation and Loading
- Data Processing and Transformation
- Index and Storage Structures
- Queries and OLAP
- Query Processing and Optimization
- Materialized Views

Lernziele :

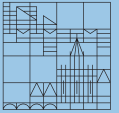
Absolventen kennen das Funktionsspektrum von Data Warehouse und OLAP-Systemen, können es von Datenbank (OLTP-) Systemen abgrenzen. Sie sind in der Lage, komplexe Datenmodelle für OLAP-Anwendungen mit den State-of-the-art Techniken zu modellieren und auf relationale DW-Systemen umzusetzen. Ferner können sie mit Hilfe von diversen grafischen Werkzeugen OLAP-Anfragen formulieren, und auch deren Formulierung in SQL bereitet keine Schwierigkeiten. In die Rolle von Entscheidern versetzt verfügen sie über die notwendigen Fachkenntnisse, um kompetente Entscheidungen über System- (DW) und Tool-Auswahl zu fällen. Die Herausforderungen beim ETL-Prozess und beim Performance-Tuning von DW-Systemen sind bekannt.

Voraussetzung :

Grundlegende Kenntnisse über Datenbanken, wie sie etwa im Modul Datenbanken der Bachelorstudiengänge Information Engineering und Informatik vermittelt werden; allgemeine Programmierkenntnisse und Verständnis von Datenstrukturen und Algorithmen

Arbeitsaufwand :

180 Stunden, davon ca. 60h in Präsenz und 120h im Selbststudium



Prüfungsleistung :

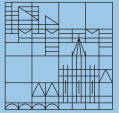
Je nach Teilnehmerzahl mündlichen oder schriftliche Prüfung (wird zu Beginn bekannt gegeben). Über den praktischen (Übungs-) Anteil kann ein Bonus für die Prüfung erworben werden.

Literatur :

Die Veranstaltung orientiert sich an dem Buch

- V. Köppen, G. Saake, K.-U. Sattler. *Data Warehouse Technologien*. mitp-Verlag, Heidelberg, 2012

Weitere Literatur wird in der Veranstaltung bekannt gegeben.



15.25 Usability Engineering: Evaluation (Usability Engineering: Evaluation)

Modulverantwortlicher : Prof. Dr. Reiterer (z.B. INF-10025-20121)

Credits : 4 SWS, 6 ECTS

Inhalte :

Evaluation ist ein integraler Bestandteil des Usability Engineering. Sie dient dazu, möglichst frühzeitig Benutzungsprobleme von in der Entwicklung befindlichen interaktiven Produkten festzustellen und konkrete Ideen für Verbesserungen zu liefern. Mittlerweile ist ein breites Spektrum an unterschiedlichsten Verfahren und Methoden verfügbar (z.B. Beobachtungen, Benutzertests, Fragebögen, etc.), die sich in Aspekten wie Zeitpunkt der Anwendung im Entwicklungsprozess, Einsatz von Experten oder Beteiligung von potentiellen Endnutzern teils deutlich unterscheiden und somit auch Antworten auf unterschiedliche Fragestellungen liefern. Im theoretischen Teil der Veranstaltung werden den Studenten mehrere dieser Verfahren eingehend vorgestellt. Hierzu gehören Grundkenntnisse bezüglich Fragebogenentwicklung, Interviews, Focus Groups, Usability Tests, Eye-Tracking, Tagebuchtechnik, Beobachtung, etc. Im Rahmen des Kurses werden zudem auch die Grundlagen des Designs von Experimenten vermittelt (z.B. wie wird eine Hypothese formuliert, was sind unabhängige und abhängige Variablen, was ist ein „within-subjects design“, wie können die erhobenen Daten statistisch ausgewertet werden, etc.). Anschließend wird im praktischen Teil ein Projekt von den Studenten selbst in kleinen Gruppen durchgeführt werden. Die Studenten erhalten dabei die Gelegenheit, ein Usability Labor zu nutzen und ihre eigenen Erfahrungen hinsichtlich der Evaluation interaktiver Produkte zu sammeln und die theoretisch erlernten Methoden konkret anzuwenden.

Lernziele :

Das Lernziel dieses Kurses besteht darin, die notwendigen Methoden und Vorgehensweisen zur Durchführung der folgenden Aktivitäten beim Entwurf von interaktiven Produkten (Prozessmodell des Interaction Design, siehe Modul Vorlesung Mensch- Computer Interaktion) im Detail zu vermitteln:

- Evaluation interaktiver Produkte hinsichtlich ihrer Gebrauchstauglichkeit
- Ableitung von Änderungserfordernissen zur Beseitigung festgestellter Fehler und Mängel.

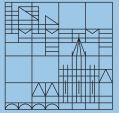
Neben einer theoretischen Auseinandersetzung mit dem Thema besteht der praktische Teil aus der konkreten Durchführung einer Evaluationsstudie.

Voraussetzung :

Die Inhalte der Vorlesung „Mensch-Computer Interaktion“ bzw. „Computergrafik und Interaktive Systeme“ sollten bekannt sein oder durch Eigenstudium (siehe Literatur) vor Beginn der Veranstaltung erarbeitet worden sein.

Arbeitsaufwand :

Insgesamt 180 Stunden, verteilt auf:



- 56 Stunden Präsenzstunden
- 124 Stunden Eigenstudium

Prüfungsleistung :

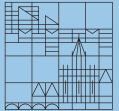
Die Prüfungsleistung besteht aus folgenden Aspekten:

- einstündige Klausur über die Inhalte des Theorieteils
- die Teilnahme an einem praktischen Projekt sowie die Bewertung von Projekt-Milestones
- Abschlusspräsentation des praktischen Projekts
- Abschlussbericht (im Format der Abschlusspräsentation)

Literatur :

Textbücher:

- Lazar et al., *Research Methods in HCI*, John Wiley & Sons, 2009
- Preece J. et al., *Interaction Design* (3rd edition), John Wiley & Sons, 2011, Chapters 12-15



15.26 Visuelle Suchsysteme (Visual Information Seeking Systems)

Modulverantwortlicher : Prof. Dr. Reiterer (z.B. INF-10190-20131)

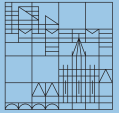
Credits : 4 SWS, 6 ECTS

Inhalte :

Heutige Informationsräume - egal ob digitale Bibliotheken, das lokale Dateisystem, Warenkataloge im Web, Wikipedia oder Businessdaten eines Konzerns - wachsen scheinbar unaufhörlich in ihrer Größe und Komplexität. Während immer schnellere Such- und Indexierverfahren den Zugriff durch herkömmliche Suchanfragen beschleunigen, werden die Benutzungsschnittstellen zur Suche, zur Trefferdarstellung und zum Browsing oft vernachlässigt. Oftmals herrschen hier noch Ansätze vor, wie sie bereits vor Jahrzehnten zum Einsatz kamen, ohne dass deren Gestaltung kritisch hinterfragt wird. Im Kurs "Visuelle Suchsysteme" wird gezeigt, wie durch den Einsatz von interaktiven Visualisierungen der Rechercheprozess für den Benutzer wesentlich effektiver, effizienter und auch unterhaltsamer gestaltet werden kann. Die Studenten lernen, wie man Benutzer eines Informationssystems mit geeigneten Visualisierungen bei der Formulierung seines Informationsbedürfnisses (Formulierung der Suchanfrage) und durch eine durchdachte Präsentation der Suchergebnisse unterstützen kann. Mit innovativen Benutzungsoberflächen, wie z.B. animierten Zoomable User Interfaces (ZUI), oder neuen Ein- und Ausgabegeräten, wie z.B. der Powerwall, verschwimmen dabei oft die traditionellen Grenzen zwischen Suchanfrage, Trefferliste und Navigation. Am Beginn des Kurses wird im Theorieteil eine Einführung in die Disziplin der interaktiven Informationsvisualisierung anhand von Theorie und von ausgewählten Beispielen gegeben (siehe auch die Module Information Visualization I und II). Dann wird detailliert auf das Spezialgebiet der "visuellen Suchsysteme" eingegangen. Illustriert werden diese anhand einer Reihe von visuellen Suchsystemen und Konzepten aus der Arbeitsgruppe Mensch-Computer-Interaktion (z.B. aus den Projekten MedioVis, ZOIL), die dann in der Veranstaltung auch von ihren jeweiligen Entwicklern demonstriert und diskutiert werden. Dabei wird auch auf den Erfolg oder Misserfolg in Benutzertests während der Usability Evaluation eingegangen. Im Praxisteil erhalten dann die Teilnehmer die Gelegenheit, ausgewählte praktische Fragestellungen im Zusammenhang mit der Entwicklung von visuellen Suchsystemen eigenständig zu bearbeiten. Hierzu wird eine neue Visualisierung oder Interaktionstechnik für eines der obengenannten Systeme implementiert und die Implementation in einem 10-15seitigen Dokument festgehalten (Hausarbeit). Die Ergebnisse werden zusätzlich vom jeweiligen Teilnehmer in einem abschließenden Kolloquium vor den anderen Kursteilnehmern in einem ca. 30minütigen Vortrag präsentiert.

Lernziele :

Das grundsätzliche Lernziel besteht darin, die Möglichkeiten des Einsatzes von Visualisierungen zur Unterstützung von Recherchen in digitalen Datenräumen (z.B. im Web, in digitalen Bibliotheken, in online-Datenbanken) kennen zu lernen. Des Weiteren soll den Teilnehmern vermittelt werden, dass ein effektiver und effizienter Rechercheprozess nur zu erreichen ist, wenn die Visualisierungen in geeignete Interaktionskonzepte eingebettet sind, die sich durch eine hohe Gebrauchstauglichkeit auszeichnen. Im einzelnen können die Lern-



ziele wie folgt formuliert werden: Die Teilnehmer erlernen die Möglichkeiten und Grenzen von Visualisierungen zur Exploration digitaler Informationsräume. Die Teilnehmer lernen verschiedene Interaktionskonzepte (z.B. Zoomable User Interfaces) und deren Anwendung auf unterschiedlichen Endgeräten (z.B. PDA, Desktop, PowerWall) kennen. Die Teilnehmer erreichen ein tieferes Verständnis über theoretische bzw. konzeptionelle Fragestellungen im Zusammenhang mit dem Einsatz von Visualisierung und deren Einbettung in geeignete Interaktionstechniken. Die Teilnehmer sind in der Lage, eigenständig eine spezielle Visualisierung in Kombination mit einem geeigneten Interaktionskonzept zu implementieren, die sich durch eine hohe Gebrauchstauglichkeit (Usability) auszeichnet

Voraussetzung :

Die Inhalte der Vorlesungen “Mensch-Computer Interaktion“ bzw. “Computergrafik und Interaktive Systeme“ sollten bekannt sein oder durch Eigenstudium (siehe Literatur) vor Beginn der Veranstaltung erarbeitet worden sein.

Arbeitsaufwand :

Insgesamt 180 Stunden, verteilt auf:

- 56 Stunden Präsenzstudium
- 124 Stunden Eigenstudium

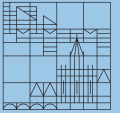
Prüfungsleistung :

Die Prüfungsleistung besteht aus folgenden Aspekten:

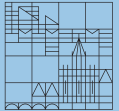
- einstündige Klausur über die Inhalte des Theorieteils (Mitte des Semesters!)
- die Teilnahme an einem praktischen Projekt sowie die Bewertung von Projekt-Milestones
- Abschlusspräsentation des praktischen Projekts
- Abschlussbericht (im Format der Abschlusspräsentation)

Literatur :

- Card, Stuart K.: Information Visualization. In: Jacko Julie A.; Sears Andrew (Eds.): *The Human-Computer Interaction Handbook*. Mahwah, NJ (Lawrence Erlbaum) 2003, pp.544-582.
- Bederson, B. B., Shneiderman, B. (Hrsg.): *The Craft of Information Visualization. Readings and Reflections*. San Francisco, CA, Morgan Kaufmann, 2003
- Card, Stuart K.; Mackinlay, Jock D.; Shneiderman, Ben (Hrsg.): *Readings in Information Visualization. Using Vision to Think*. San Francisco, CA (Morgan Kaufmann Publishers, Inc.) 1999
- Spence, Robert: *Information Visualization*, Addison-Wesley, 2001
- Reiterer, H.: Visuelle Exploration digitaler Datenbestände. In: Maximilian Eibl; Harald Reiterer; Peter F. Stephan; Frank Thissen (Eds.): *Knowledge Media Design. Theorie, Methodik, Praxis.*, Oldenbourg, München, 2005, siehe: http://hci.uni-konstanz.de/downloads/Reiterer_VisuelleExplorationDigitalerDatenbestaende.pdf



- Hearst, Marti: User Interfaces and Visualization, in: Ricardo Baeza-Yates, Berthir Ribeiro-Neto; *Modern Information Retrieval*, acm Press, 1999
- Gerken, Jens; Heilig, Mathias; Jetter, Hans-Christian; Rexhausen, Sebastian; Demarmels, Mischa; König, Werner A.; Reiterer, Harald: Lessons Learned from the Design and Evaluation of Visual Information Seeking Systems. in: Adam, Nabil; Furuta, Richard; Neuhold, Erich, Springer, *International Journal on Digital Libraries*, p. 49–66, Aug 2009



15.27 Usability Engineering: Design (Usability Engineering: Design)

Modulverantwortlicher : Prof. Dr. Reiterer (z.B. INF-10960-20122)

Credits : 4 SWS, 6 ECTS

Inhalte :

Inhalt des Kurses ist die Vermittlung einer systematischen Vorgehensweise bei der Entwicklung gebrauchstauglicher und benutzerzentrierter interaktiver Produkte (z.B. Anwendungssoftware, Webseiten, Information Appliances etc.). Dabei stehen die Vermittlung, der Vergleich und die praktische Anwendung verschiedener in der Literatur vorhandener Vorgehensmodelle für das benutzerzentrierte Design im Zentrum des Kurses. Einen praktischen Schwerpunkt stellt dabei die Ausarbeitung von innovativen Produktkonzepten und Designs dar. Neben einer theoretischen Auseinandersetzung mit dem Thema erfolgt eine intensive praktische Umsetzung der Inhalte im Rahmen der Entwicklung von Designstudien für interaktive Produkte.

Die Inhalte des Kurses basieren hauptsächlich auf vier gängigen Vorgehensmodellen des Usability Engineering (siehe Literatur) mit Einbezug einer explorativen Design-Phase. Während des Kurses werden die Inhalte dieser Vorgehensmodelle vermittelt. Gemeinsamkeiten und Unterschiede der Vorgehensmodelle werden dabei im direkten Vergleich deutlich und individuelle Vor- und Nachteile werden herausgearbeitet. Der Schwerpunkt liegt dabei auf folgenden Inhalten:

- Generelle Prozessmodelle und Vorgehensweisen im Usability Engineering
- Kreative Techniken zur Exploration von Designproblemen und Designlösungen (Affinity Diagramming, Sketching)
- Methoden zur Erfassung der Bedürfnisse und Anforderungen des Nutzungskontextes (z.B. Benutzerrollen, Aufgabenanalysen)
- Methoden zur Erstellung von Designstudien (z.B. Konzeptionelles und physisches Interaction Design, Prototyping)

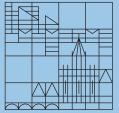
Im praktischen Teil des Kurses werden dann in Gruppen diese Modelle auch praktisch angewendet, um Ideen und Konzepte zu entwickeln und in ansprechenden Designstudien den anderen Kursteilnehmern zur Diskussion und zur abschließenden Bewertung vorzustellen.

Lernziele :

- Erfassen von Bedürfnissen und Ableiten von Anforderungen aus dem Nutzungskontext
- Entwickeln von konzeptionellen Modellen und Designalternativen (Modelling & Sketching)
- Entwicklung von interaktiven Designstudien (Mockups & Prototypen)

Voraussetzung :

Die Inhalte der Vorlesung "Computergrafik und Interaktive Systeme" sollten bekannt sein oder durch Eigenstudium (siehe Literatur) vor Beginn der Veranstaltung erarbeitet worden sein.

**Arbeitsaufwand :**

Insgesamt 180 Stunden, verteilt auf:

- 56 Stunden Präsenzstudium
- 124 Stunden Eigenstudium

Prüfungsleistung :

Einstündige schriftliche Klausur über die Inhalte des Theorieteils. (Mitte des Semesters!) Durchführung eines benutzerzentrierten Designprozesses in der Gruppe unter Befolgung eines Vorgehensmodells. Dokumentation aller Arbeitsschritte und Zwischenergebnisse. Regelmäßige Präsentation der entstandenen Zwischenergebnisse anhand von anschaulichen Diagrammen, ansprechenden Skizzen und interaktiven Prototypen. Präsentationstermine: Mitte bis Ende der VL-Zeit. Anwesenheit an diesen Terminen ist für alle Kursteilnehmer verpflichtend! Als Ausarbeitung wird eine Dokumentation des Projektablaufs anhand von Zwischenpräsentationen, Artefakten, Modellen, Styleguides etc. im BSCW erwartet. Der Projektablauf und Designentscheidungen müssen an der Dokumentation nachvollzogen werden können und in einem zusätzlichen Dokument erläutert werden (z.B. Foliensatz).

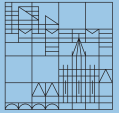
Literatur :

Als Einstieg in das Thema wird empfohlen (sofern die Inhalte des Kurses Mensch-Computer Interaktion nicht bekannt sein sollten):

- Preece J., Rogers Y., Sharp H., *Interaction Design*, John Wiley & Sons, 2002

Die Inhalte des Kurses basieren auf folgenden Lehrbüchern:

- Buxton B., *Sketching User Experiences: Getting the Design Right and the Right Design*, Morgan Kaufmann, 2007
- Constantine L., Lockwood L., *Software for use*, Addison-Wesley, 1999
- Mayhew D., *The Usability Engineering Lifecycle*, Morgan Kaufmann, 1998
- Rosson M., Carroll J., *Usability Engineering - Scenario-based Development of Human-Computer Interaction*, Morgan Kaufmann, 2002
- Beyer H., Holtzblatt K., *Contextual Design*, Morgan Kaufmann, 1998



15.28 Camera Calibration for Computer Vision

Modulverantwortlicher : Prof. Dr. Saupe (z.B. INF-11750-20112)

Credits : 6 SWS, 9 ECTS

Inhalte :

Kamerakalibrierung bezeichnet den Vorgang, zu einem Bild oder einer Bildsequenz die Parameter eines mathematischen Modells der Kamera zu bestimmen, die die Aufnahmen gemacht hat. Das Modell ist eine 3×4 Kameramatrix, welche die projektive Abbildung der 3D Szene auf die Pixelkoordinaten der Kamera definiert. Kamerakalibrierung ist grundlegend für Computersehen und Augmented Reality. In den Vorlesungen behandeln wir die theoretischen Grundlagen, Algorithmen und Anwendungen. In den Übungen werden mehrere Kalibrierungsalgorithmen für virtuelle und reale Kameras implementiert. Die Verfahren werden für einfache praktische Aufgaben eingesetzt, z.B. um die Distanz zwischen zwei 3D Punkten aus Stereobildpaaren abzuleiten.

Lernziele :

Die Studierenden

- verstehen die Konzepte der Abbildung von Welt- auf Kamerakoordinaten und sind in der Lage, mathematisch die 2D projektiven Transformationen anzugeben
- können die Methoden der 2D projektiven Geometrie auf 2D Transformationen anwenden
- verstehen es, ein mathematisches Modell für Bilder einer realen Kamera zu bestimmen
- sind in der Lage, grundlegende Techniken des Computersehens für Kamerakalibrierung in MATLAB Programmen zu implementieren

Voraussetzung :

Keine. Empfohlen wird, zuvor eine einführende Vorlesung in Computergrafik gehört zu haben. Falls der Kurs im Fortgeschrittenenstudium eines der Bachelorstudiengänge belegt wird:

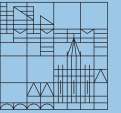
- Mathematikurse des Bachelorstudiengangs
- Algorithmen und Datenstrukturen
- einführende Kurse in Informatik und Programmieren

Arbeitsaufwand :

- ECTS: 9 Punkte
- SWS: 3 Stunden Vorlesung, 3 Stunden Übung
- Insgesamt: 270 Stunden, davon 84 Stunden Präsenzstudium

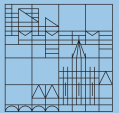
Prüfungsleistung :

Übungsaufgaben, Klausur



Literatur :

- Richard Hartley, Andrew Zisserman, *Multiple View Geometry in Computer Vision*, Second Edition, Cambridge, 2003.



15.29 Digitale Signalverarbeitung (Digital Signal Processing)

Modulverantwortlicher : Prof. Dr. Saupe (z.B. INF-10150-20122)

Credits : 6 SWS, 9 ECTS

Inhalte :

Dieser einführende Kurs in Digitaler Signalverarbeitung hat folgende Inhalte:

- Zeitdiskrete Signale und Systeme
- FIR und IIR Filter
- Zeitdiskrete Fouriertransformation
- Abtastung und Interpolation von Signalen
- Frequenzraumanalyse und schnelle Fouriertransformation (FFT)

Lernziele :

Die Studierenden

- kennen und beherrschen die grundlegenden Konzepte der digitalen Signalverarbeitung
- erlangen ein vertieftes Verständnis der Konzepte und Methoden der zeitkontinuierlichen und zeitdiskreten Fouriertransformationen
- beherrschen die Frequenzraum-Analyse zeitdiskreter Signale und Systeme
- haben Erfahrungen in der Praxis der digitalen Signalverarbeitung unter Verwendung von MATLAB oder anderen Programmiersprachen

Voraussetzung :

Keine, wenn der Kurs im Rahmen des Masterstudiengangs Information Engineering belegt wird. Als Kurs im Fortgeschrittenstudium im Bachelorstudiengang Information Engineering oder Informatik:

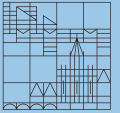
- Mathematikurse des Bachelorstudiengangs
- Algorithmen und Datenstrukturen
- einführende Kurse in Informatik und Programmieren

Arbeitsaufwand :

- ECTS: 9 Punkte
- SWS: 4 Stunden Vorlesung, 2 Stunden Übung
- Insgesamt: 270 Stunden, davon 84 Stunden Präsenzstudium

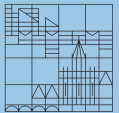
Prüfungsleistung :

Übungsaufgaben und Klausur



Literatur :

- Alan Oppenheimer, Ronald Schafer, John Buck, *Discrete-time Signal Processing*, Prentice-Hall, 2010.
- Alan Oppenheimer, Ronald Schafer, John Buck, *Zeitdiskrete Signalverarbeitung*, 2. Auflage, Pearson Studium, 2004.
- Vinay Ingle, John Proakis, *Digital Signal Processing using MATLAB*, Third Edition, Cengage Learning, 2012.



15.30 Architektur und Implementierung von Datenbanksystemen (Database System Architecture and Implementation)

Modulverantwortliche : Jun.-Prof. Dr. Grossniklaus, Prof. Dr. Scholl (z.B. INF-10555-20102)

Credits : 4 SWS, 6 ECTS

Inhalte :

Modern database systems use advanced data structures and algorithms to manage and process data. Building on the basic database system course, this course looks at the internals of database systems in terms of their architecture and implementation. The lectures are structured according to the levels of a general layered software architecture for database systems, which is introduced at the beginning of the course. The course starts at the lowest level with the I/O system and the management of buffers and storage, i.e., disk and file management. Moving up a level of the architecture, the course then introduces different kinds of index structures, such as tree-based and hash-based indexes. Continuing from this groundwork, the course then focuses on query processing in terms of query execution and optimization. Finally, the course will teach how benchmarks can be used to qualitatively and quantitatively assess the performance of different database systems. Gathering and understanding results of database system benchmarks is the basis for selecting and tuning a database system for a given application. Throughout the course, students have a chance to get first-hand practical experience with the presented techniques and implementation concepts through a series of mandatory programming exercises.

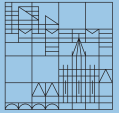
Lernziele :

In this course, students will learn to understand and apply the data structures and algorithms that are used to realize the storage, indexing, and query processing functionality of contemporary relational database systems. In doing so, students will acquire the ability to differentiate between operating and database system functionality. Finally, the course imparts the knowledge required to assess the qualitative and quantitative differences between different relational database systems. Therefore, the course syllabus will enable students to make informed choices when selecting or tuning a database system for a given application.

Voraussetzung :

The following skills, knowledge and courses are mandatory prerequisites to attend and successfully complete this course.

- **Basics of database systems:** database design, query languages, database application programming, etc. (INF-12040 or equivalent)
- **Principles of database systems:** relational model, relational algebra, normal forms, etc. (INF-12040 or equivalent)
- **Computer systems:** computer architecture, operating systems, networks, etc. (INF-11740, INF-11880, or equivalent)
- **System programming:** students must have the ability to program in a language appropriate for system implementation, such as C/C++, C# or Java. (INF-11930 or equivalent)



- **Key competences:** Subversion, LaTeX, etc. (INF-10175 or equivalent)

Arbeitsaufwand :

180 hours, of which ca. 60 hours are spent in class and another 120 hours are spent on exercises and project work.

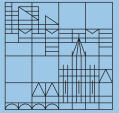
Prüfungsleistung :

The final exam of this course will be conducted as a 2-hour written exam. The exam will be weighted 50% in the final grade of this course with exercises and project work contributing another 50% of the grade.

Literatur :

The following study materials are relevant for this course.

- **Textbook:** Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems (3rd Edition), McGraw-Hill, 2002
- **Course slides:** Copies of the slide decks used in the course will be made available online
- **Reading list:** Publications referenced in the course as well as further readings will be available on the course website



15.31 Transaktionsorientierte Informationssysteme (Transactional Information Systems)

Modulverantwortlicher : Prof. Dr. Scholl (z.B. INF-11610-20121)

Credits : 4 SWS, 6 ECTS

Inhalte :

Transaktionssysteme sind eine Schlüsseltechnologie für zentralisierte und verteilte Mehrbenutzersysteme. Sie gewährleisten, dass parallele Prozesse aus der Sicht des Anwenders und des Anwendungsprogrammierers atomar, persistent und untereinander isoliert ablaufen. Diese generischen Systemdienste sind eine fundamentale Vereinfachung für die Entwicklung von mehrbenutzerfähigen und fehlertoleranten Anwendungen, bei denen es auf Datenkonsistenz und Verlässlichkeit ankommt. Aus diesem Grund sind Transaktionen eines der wichtigsten Konzepte in Datenbanksystemen. Aber auch im Bereich der Betriebssysteme und Programmiersprachen gewinnen Transaktionskonzepte zunehmend an Bedeutung. Transaktionstechnologie ist darüber hinaus auch einer der Eckpfeiler moderner Internet-Anwendungen wie Electronic-Commerce oder Workflow-Management. Die Vorlesung behandelt sowohl theoretische Grundlagen als auch Implementierungstechniken für leistungsfähige Transaktionsprotokolle ("concurrency control protocols") und Möglichkeiten der Fehlerbehandlung ("recovery") von Transaktions-, Systems- und Medienfehlern in zentralisierten und verteilten Systemen.

Lernziele :

Absolventen können verschiedene Arten von transaktionsorientierter Verarbeitung voneinander abgrenzen und vergleichend bewerten. Sie sind in der Lage, Transaktionssysteme bzgl. ihrer detaillierten Funktionalität, ihrer Performancepotenziale und -engpässe zu evaluieren und für konkrete Anwendungszwecke auszuwählen.

Voraussetzung :

Grundverständnis für die Funktionsweise von komplexen Anwendungssystemen, sowie grundlegende Kenntnisse aus den Bereichen Datenstrukturen und Algorithmen sowie Methoden der Praktischen Informatik. Der vorherige Besuch der Veranstaltung Datenbanken (Bachelorstudiengang Information Engineering oder Informatik) ist empfehlenswert, wird aber nicht zwingend vorausgesetzt.

Arbeitsaufwand :

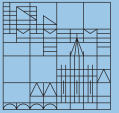
180 Stunden, davon ca. 60 Stunden in Präsenz und 120 Stunden Selbststudium

Prüfungsleistung :

Je nach Teilnehmerzahl entweder eine schriftliche Klausur von ca. 120min Dauer oder eine mündliche Prüfung von ca. 30min im Prüfungszeitraum des jeweiligen Semesters

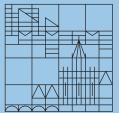
Literatur :

Zur Vorlesung steht ein begleitendes Folienskript zur Verfügung. Die Veranstaltung orientiert sich an dem Buch *Transactional Information Systems: Theory, Algorithms, and the Practice of Concurrency Control and Recovery* von Gerhard Weikum und Gottfried Vossen, Morgan



Kaufmann Publishers, 2001. Als weiterführende bzw. vertiefende Literatur sind die im obigen Buch genannten Aufsätze sowie folgende Bücher geeignet:

- P. A. Bernstein, V. Hadzilacos, N. Goodman: *Concurrency Control and Recovery in Database Systems*, Addison-Wesley Publishing Company, 1987.
- A. K. Elmagarmid: *Database Transaction Models for Advances Applications*, Morgan Kaufmann Publishers, 1992.
- J. Gray, A. Reuter: *Transaction Processing: Concepts and Technologies*, Morgan Kaufmann Publishers, 1993.
- V. Kumar, M. Hsu: *Recovery Mechanisms in Database Systems*, Prentice Hall, 1998.



15.32 XML und Datenbanken (Database Supported XML)

Modulverantwortlicher : Prof. Dr. Scholl (z.B. INF-2441-20092)

Credits : 6 SWS, 9 ECTS

Inhalte :

Die Teilnehmer dieser Vorlesung werden wesentliche Techniken zur Einbindung von XML-strukturierten Daten in Datenbanksysteme kennenlernen und teilweise auch an Übungsaufgaben ausprobieren. Schwerpunkte sind u. a.:

- Codierungsverfahren, um XML-Daten in relationalen Datenbanken abzulegen
- Anfrageformulierung und -auswertung auf XML-Dokumenten
- Indizierung von XML-Daten
- native XML-Datenbanken, die unmittelbar auf XML-Daten arbeiten
- andere “NoSQL“-Datenbanksysteme

Lernziele :

Absolventen können verschiedene Systeme bzgl. des Grades und der Art der Unterstützung des XML-Datenformates analysieren und bewerten. Sie verfügen über detaillierte Kenntnisse der W3C-Sprachen rund um das XML-Format (z.B. XPath und XQuery), können diese Sprachen zur Transformation von XML-Daten und zur Abfrage von XML-Datenbanken anwenden. Ferner sind sie in der Lage, verschiedene Realisierungstechniken (Datenstrukturen und Algorithmen) von XML-Datenbanken zu verstehen, zu bewerten und ggf. zwischen ihnen in Abhängigkeit vom Anwendungsprofil auszuwählen. Bei der Auswahl möglicher “Produkte“ (kommerziell oder frei) können diese Kenntnisse für fundierte Entscheidungen herangezogen werden.

Voraussetzung :

Grundlegende Kenntnisse über Datenbanksysteme, wie sie etwa im Modul Datenbanken der Bachelorstudiengänge Information Engineering oder Informatik vermittelt werden; allgemeine Programmierkenntnisse; Vorkenntnisse über XML sind hilfreich, aber nicht notwendig

Arbeitsaufwand :

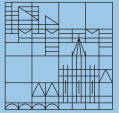
270 Stunden, davon ca. 90 Stunden in Präsenz und 180 Stunden im Selbststudium.

Prüfungsleistung :

Je nach Teilnehmerzahl eine schriftliche Klausur von ca. 90min Dauer oder eine mündliche Prüfung von etwa 30min, jeweils im Prüfungszeitraum des Semesters.

Literatur :

Begleitend zur Veranstaltung werden den Teilnehmern Originalarbeiten zu den einzelnen Themenbereichen zur Verfügung gestellt. Der Themenbereich als ganzes ist derzeit in Form von Lehrbüchern oder Überblicksarbeiten nicht aufbereitet.



15.33 Information Visualization I & II

Modulverantwortlicher :

Prof. Dr. Keim (z.B. INF-12730-20131)

Credits :

3 SWS, 4 ECTS (Information Visualization I)

3 SWS, 5 ECTS (Information Visualization II)

Inhalte :

Der Doppelkurs "Information Visualization I und II" gibt einen weitreichenden Einblick in die Forschungsergebnisse auf dem Gebiet Informationsvisualisierung der letzten Jahre. Der Kurs teilt sich in zwei Teile auf, die aufeinander aufbauen und die je nach Voraussetzung einzeln oder zusammen belegt werden können. Der erste Teil (2+1 SWS, 4 ECTS) führt in die Grundlagen der Informationsvisualisierung ein und kann von allen Teilnehmern besucht werden, die noch keine Lehrveranstaltung in Informationsvisualisierung gehört haben, z.B. innerhalb der Vorlesung Analyse und Visualisierung. Er wird mit einer Klausur abgeschlossen. Der zweite Teil (2+1 SWS, 5 ECTS) führt in erweiterte Techniken und Probleme der Informationsvisualisierung ein. Er wird von praktischen Übungen begleitet und voraussichtlich mit einem Kurzkolloquium abgeschlossen. Die Veranstaltung deckt relevante Aspekte der menschlichen Wahrnehmung, Grundlagen des Designs von visuellen Abbildungen, Anwendungen der Informationsvisualisierung sowie relevante Interaktionsmöglichkeiten ab. Es werden Methoden der Informationsvisualisierung für unterschiedliche Datenarten, z.B. 1-, 2-, und 3-dimensionale, hierarchische und räumliche Daten behandelt. Des Weiteren werden aktuelle Themen wie Visualisierungen für jedermann oder „Streaming Visualisierungen“ diskutiert.

Lernziele :

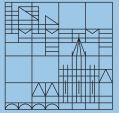
- Die Studierenden verstehen die Notwendigkeit von Informationsvisualisierungen.
- Die Studierenden sind in der Lage große unbekannte Datenmengen aufzuarbeiten, zu analysieren und angemessen darzustellen.
- Die Studierenden sind in der Lage unterschiedliche Informationsvisualisierungen kritisch zu bewerten und systematisch Informationsvisualisierungssysteme für neue Anwendungen zu entwerfen.

Darüber hinaus für Teil 2:

- Die Studierenden verstehen fortgeschrittene Konzepte der Informationsvisualisierung und der visuellen Datenanalyse, einschließlich aktueller Fragestellungen in der Forschung.
- Die Studierenden erlernen in Gruppenarbeit eine Informationsvisualisierung zu implementieren und auf eine reale Fragestellung anzuwenden.

Voraussetzung :

- Grundlegende Programmierkenntnisse



- Grundlegende Kenntnisse über Datenbanken und Anfragesprachen
- Besuch der Vorlesungen Datenbanksysteme, Modul Informatik 1 & 2 ist verpflichtend

Arbeitsaufwand :

Insgesamt 126 Stunden, verteilt auf

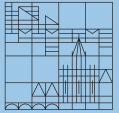
- 42 Stunden Präsenzstudium (4 SWS Vorlesung * 7 Wochen + 2 SWS Übung * 7 Wochen)
- 84 Stunden Eigenstudium
- 4 Credits

Prüfungsleistung :

mündliche oder schriftliche Prüfung

Literatur :

- Ward, M., Grinstein, G. and Keim, D. *Interactive Data Visualization: Foundations, Techniques, and Applications*, 2010 ISBN 9781568814735
- Ware, C. *Information Visualization: Perception for Design*, Morgan Kaufmann, 2nd edition, 2004
- Andrienko, N. and Andrienko, G. *Exploratory Analysis of Spatial and Temporal Data, A Systematic Approach*, Springer, 2006
- MacEachren, A. M., *How Maps Work: Representation, Visualization, and Design*, The Guilford Press, 2004
- Aigner, W., Miksch, S., Schumann, S. and Tominski, C., *Visualization of Time-Oriented Data*, Human-Computer Interaction Series, Springer, 2011



15.34 Design Patterns and Concurrency (Design Patterns and Concurrency)

Modulverantwortlicher : Prof. Dr. Waldvogel (z.B. INF-11530-20121)

Credits : 2 SWS, 3 ECTS

Inhalte :

This lecture is about Design Patterns and Concurrency. Based on Java as main programming language we examine what Design Patterns can offer and how Concurrency increases performance related to Java programs.

Voraussetzung :

- Valid registration over LSF (first come, first serve; only 15 seats; acceptance per Mail)
- student in Master Course
- extended knowledge in Java and Object-Oriented Paradigms

Arbeitsaufwand :

Präsenzstudium: 28 Stunden, Eigenstudium: 62 Stunden

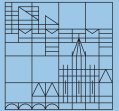
Literatur :

Concurrency:

- Brian Goetz et al., *Java Concurrency in Practice*, Addison-Wesley, 2006, ISBN-13 978-0-321-34960-6.

Design Patterns:

- Erich Gamma et al., *Design Patterns: Elements of Reusable Object-Oriented Software 1*, Addison-Wesley, 1994, ISBN 78-0201633610.
- Elisabeth Freeman et al., *Head First Design Patterns*, O'Reilly, 2004, ISBN 978-0596007126.



15.35 Directed Studies: Advanced Topics in Software and Systems Engineering

Modulverantwortlicher : Prof. Dr. Leue (z.B. INF-12710-20131)

Credits : 3 SWS, 5 ECTS

Inhalte :

This directed studies course will introduce into the areas of embedded systems and functional safety engineering for those systems. Selected topics from these areas will be discussed and presented by participants on a weekly basis. Lectures will be accompanied by a tutorial.

Lernziele :

Participants of this course will be enabled to understand and assess methods used in the design of safe embedded systems. This will enable them to employ adequate techniques in practical system engineering projects. Participants will also learn to acquire technical knowledge from textbook sources and to present this knowledge to their peers.

Voraussetzung :

Bachelor level knowledge of operating systems, distributed systems, software engineering and theory.

Arbeitsaufwand :

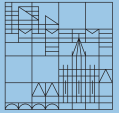
150 hrs. (5 ECTS)

Prüfungsleistung :

Final examination. At least one presentation of a course topic (appr. 30 mins.).

Literatur :

Literature will be announced during the course.



15.36 Introduction to Information Theory (Introduction to Information Theory)

Modulverantwortlicher : Prof. Dr. Saupe (z.B. INF-12520-20121)

Credits : 4 SWS, 6 ECTS

Inhalte :

Informationstheorie ist die Wissenschaft grundlegender Eigenschaften von Daten hinsichtlich Speicherung, Kompression und Kommunikation. Der Kurs beginnt mit dem Begriff der Entropie und führt die variable Längencodierung ein. Der erste Fundamentalsatz von Claude Shannon (noiseless coding theorem) stellt den Zusammenhang zwischen der mittleren Codewortlänge einer Quellcodierung und der Entropie her. Bei der Übertragung von Information über einen gestörten Kanal sind jedoch andere Codierungen notwendig, um Effizienz zu maximieren und zusätzlich die Wahrscheinlichkeit von unkorrigierbaren Übertragungsfehlern zu minimieren. Das führt zum Begriff der Kapazität eines Kanals und dem zweiten Hauptsatz von Shannon. Quantifizierungen zu Information und Unsicherheit wie z.B. Entropie, relative Entropie und Transinformation haben vielfältige Anwendungen in z.B. Mustererkennung, Computersehen und Plagiarismuserkennung.

Lernziele :

Die Studierenden verstehen die Grundlagen der Informationstheorie sowie ihre Folgerungen und können sie an einfachen Beispielen anwenden.

Voraussetzung :

Elementare Stochastik, Analysis und lineare Algebra

Arbeitsaufwand :

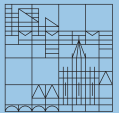
- ECTS: 6 Punkte
- SWS: 2 Stunden Vorlesung, 2 Stunden Übung
- Insgesamt: 180 Stunden, davon 56 Stunden Präsenzstudium

Prüfungsleistung :

Übungsaufgaben, Klausur

Literatur :

- S. Roman, *Coding and Information Theory, Graduate Texts in Mathematics*, Springer, 1994
- T. M. Cover and J. A. Thomas, *Elements of Information Theory, Second Edition*, Wiley, 2006.



15.37 Time Series Modeling (Time Series Modeling)

Modulverantwortlicher :

Prof. Dr. Saupe

Credits :

6 SWS, 9 ECTS

Inhalte :

Der Kurs beinhaltet einiger der klassischen mathematischen Modelle von Zeitreihen die für die Repräsentierung dynamischer Daten und deren Vorhersage verwendet werden. Im Detail: Deduktion in probabilistischen Modellen (probabilistische Inferenz, Graphkonzepte und Modelle, Belief Networks, Inferenz in Baumstrukturen) Dynamische Modelle (Markov-Modelle mit diskreten und kontinuierlichen Zuständen, alternierende lineare dynamische Systeme, verteilte Rechenmodelle in Netzwerken)

Lernziele :

Studenten habe ein grundlegendes Verständnis für die wichtigsten Methoden der probabilistischen Modellierung von Zeitreihen. Sie sind in der Lage, am Rechner für einfache Fallbeispiele geeignete Modelle und deren Parameter praktisch zu bestimmen.

Voraussetzung :

Grundlegende Mathematikkenntnisse inklusive elementarer Stochastik und Programmierkenntnisse werden vorausgesetzt.

Arbeitsaufwand :

- ECTS: 9 Punkte
- SWS: 3h Vorlesung, 3h Übung.
- Insgesamt: 270 Stunden, davon 84 Stunden Präsenzstudium.

Prüfungsleistung :

Übungsaufgaben, Klausur.

Literatur :

- David Barber, Bayesian Reasoning and Machine Learning, Cambridge University Press, 2012.