



Fakultät Informatik

Studiengang Informatik (Bachelor)

Modulhandbuch

Stand: WS 2025/26

Juli 2025

Prof. Dr. E. Müller
Studiendekan der Fakultät Informatik

Prof. Dr. E. Böhler
Studiengangkoordinator

Prof. Dr. U. Göhner
Vorsitzender der Prüfungskommission

Inhaltsverzeichnis

1 Ziele und Aufbau des Studiengangs Informatik	2
2 Begriffserläuterungen	6
3 Modulbeschreibungen	8
IFB1101 Einführung in die Informatik	8
IFB1102 Analysis	9
IFB1103 Programmieren 1	11
IFB1104 Lineare Algebra und Analytische Geometrie	13
IFB1105 Programmieren 2	15
IFB1106 Algorithmen und Datenstrukturen	17
IFB1107 Theoretische Informatik	19
IFB1108 IT-Systeme	21
IFB1109 Rechnerarchitektur	22
IFB1110 Grundlagen der Wirtschaftsinformatik	24
IFB1111 Allgemeinwissenschaftliches Wahlpflichtfach	26
IFB1112 Diskrete Mathematik	27
IFB1113 Datenbanken	29
IFB1114 Softwaretechnik 1 / Software Engineering	31
IFB1115 Betriebssysteme	33
IFB1116 Internettechnologien	35
IFB1117 Human Computer Interaction / Softwaretechnik 2	37
IFB1118 Compiler	39
IFB1119 Rechnernetze	41
IFB1120 Verteilte Softwaresysteme	43
IFB1121 Wahrscheinlichkeitsrechnung und Numerik	44
IFB1122 Software Praktikum	46
IFB1123 Projektmanagement / IT-Projektmanagement	47
IFB1124 IT-Sicherheit	49
IFB2101 Architektur und Betrieb kommerzieller Anwendungssysteme	50
IFB2102 Mikrocomputertechnik mit Praktikum	52
IFB2104 Administration von Rechnernetzen	54
IFB2106 Logik	56
IFB2107 Softwareentwicklung für Smartphones	58
IFB2109 Dokumentation und Informationssysteme	60
IFB2110 Digitale Medien	62
IFB2115 Operations Research	64
IFB2125 IT-Management	66
IFB2129 Grundlagen der Digitalen Produktion	68
IFB2130 Grundlagen von eHealth	70
IFB2131 Text Mining und Information Extraction	72
IFB2132 Advanced Embedded Systems	74
IFB2133 Telemedizin	76
IFB2135 Einführung in die Bioinformatik	78
IFB2349 Maschinelles Lernen	81
IFB2766 Funktionale Sicherheit	83
IFB2767 Einführung in Data Science	85
IFB3100.1 Praktisches Studiensemester	86
IFB3100.2A Praxisbegleitende Lehrveranstaltung Kommunizieren und Präsentieren	87
IFB3100.2B Praxisbegleitende Lehrveranstaltung Soziale Kompetenz	88
IFB4100 Seminar	89
IFB5100 Projektarbeit	90
IFB6100.1 Bachelorarbeit	91

IFB6100.2 Bachelorarbeit - Seminar.....	92
---	----

1 Ziele und Aufbau des Studiengangs Informatik

Ziele des Studiengangs Informatik sind die Vermittlung verschiedenster Kompetenzen und Lehrinhalte (vgl. hierzu auch die ausführliche Darstellung im Kapitel 2.2.1). Den Studierenden soll u.a. die Befähigung vermittelt werden, aus konkreten Fragestellungen der Praxis entstandene informationstechnische Probleme systemgerecht zu analysieren, um auf dieser Grundlage eine computerbasierte Lösung zu erarbeiten. Ziel des Studiums ist es ferner, die Studierenden zur selbständigen Anwendung wissenschaftlicher Erkenntnisse und Verfahren in dem beruflichen Feld der Informatik zu befähigen und zu qualifizieren. Das Studium vermittelt Kenntnisse, die für den Entwurf, die Implementierung und den Betrieb von komplexen informationsverarbeitenden Systemen in unterschiedlichen Anwendungsfeldern erforderlich sind. Das Informatikstudium fördert zudem die für die berufliche Praxis notwendige Fähigkeit zur Kommunikation und Teamarbeit sowie das Verantwortungsbewusstsein für den Umgang mit moderner Informationstechnik.

Der Bachelorstudiengang Informatik ist auch eine Basis und Zugangsmöglichkeit für eine anwendungsorientierte Weiterqualifizierung in den sich anschließenden Masterstudiengängen der Informatik.

Absolventen des Studiengangs Informatik erwerben während Ihres Studiums vielfältige Kenntnisse und Fähigkeiten. Sie besitzen grundlegendes Verständnis für zentrale Konzepte der Informatik, für Hard- und Softwaresysteme, sowie Kenntnisse zu den wichtigsten Informatiksystemen, wie Betriebs-, Datenbank- und Kommunikationssystemen. Zu ihren Kompetenzen zählen die Befähigung, aus konkreten Fragestellungen der Praxis entstandene informationstechnische Probleme systemgerecht zu analysieren und Lösungen unter Beachtung technischer, ökonomischer und ergonomischer Randbedingungen zu erstellen. Die Absolventen beherrschen rechnerorientierte Arbeits- und Verfahrensweisen, deren Kernpunkt die Softwareentwicklung darstellt. Sie verfügen über logisches und algorithmisches Denken und die Fähigkeit, in abstrakten Modellen zu denken. Sie besitzen Verständnis der Methodik der Modellbildung, die Fähigkeit zur Planung und Durchführung von Softwareprojekten, sind kontaktfähig und teamfähig.

Nachfolgende Tabelle zeigt die angestrebten Studienziele und Lernergebnisse des Studiengangs Informatik der Hochschule Kempten:

Nr.	Studienziel	Lernergebnisse
1	Grundlagenkompetenz	Grundlegendes Verständnis für zentrale Konzepte der Informatik Kenntnisse über formale, algorithmische und mathematische Hilfsmittel der Informatik
2	IT-Systemkompetenz	Verständnis für Hard- und Softwaresysteme Kenntnisse über Aufbau und Funktionsweise von Rechnern sowie wichtigen Informatiksystemen, wie Betriebs-, Datenbank- und Kommunikationssystemen
3	Analyse-, Design- und Realisierungskompetenz	Kenntnisse der für die Informatik typischen Vorgehensmodelle und Methoden zur Analyse, Modellierung, Realisierung und Test Fähigkeit, in abstrakten Modellen zu denken und konstruktiv vorzugehen Fähigkeit zur Lösung von Anwendungsproblemen unter Beachtung technischer, ökonomischer und ergonomischer Randbedingungen

4	Anwendungskompetenz	Kenntnisse über Aufbau von Informatiksystemen in typischen Anwendungsbereichen und Verständnis für die anwendungsbezogenen Zusammenhänge Fähigkeit, die Lösungen für spezielle Anwendungsgebiete zu erarbeiten und zu beurteilen
5	Soziale und überfachliche Kompetenzen	Kenntnisse in Arbeits-, Präsentations- und Kommunikationstechniken Fertigkeiten im Umgang mit Personen, Gruppen und Institutionen im Kontext des späteren Berufsfeldes Fähigkeit, im Team fachlich als auch leitend verantwortliche Funktionen zu übernehmen Grundlegendes Verständnis für gesellschaftliche Auswirkungen und Sozialverträglichkeit von Lösungen und Innovationen

Das Studium der Informatik teilt sich in ein Basisstudium sowie ein Vertiefungsstudium. Die Module des Basisstudiums orientieren sich an den Grundlagen der obengenannten Fachdisziplinen. Sie sollen den Studierenden zu Beginn des Studiums ermöglichen, sich in Fachdidaktik und "Fachsprachen" einzuarbeiten. Im Basisstudium finden sich folglich die Modulbereiche für die fachlichen Grundlagen in den Bereichen Mathematik, Theorie der Informatik, Software- und Computertechnik. Inhaltlich steht die "klassische Lehre" im Vordergrund. Im Vertiefungsstudium werden darüber hinaus unterschiedliche Lehr- und Lernformen eingesetzt. Es finden sich neben Seminaren und Übungen auch Projektarbeiten.

Der Zusammenhang zwischen den übergeordneten Studienzielen (1) Grundlagenkompetenz, (2) IT-Systemkompetenz, (3) Analyse- Design und Realisierungskompetenz, (4) Anwendungskompetenz und (5) Soziale und überfachliche Kompetenzen sowie den Lernergebnissen des Bachelorstudiengangs Informatik nebst dem Beitrag der Wahlpflichtmodule zur Umsetzung dieser Ziele sind in der folgenden Zielematrix dargestellt:

Modul	Studienziele				
	1	2	3	4	5
Algorithmen und Datenstrukturen	++			+	
Allgemein Wissenschaftliches Wahlpflichtfach					++
Analysis	++		+		
Bachelorarbeit				++	+
Betriebssysteme	+	++			
Compiler	+	+		+	
Datenbanken		++			
Diskrete Mathematik	++		+		
Einführung in die Informatik	++				+
Fachwissenschaftliche Wahlpflichtfächer		+	+	+	+
Grundlagen der Wirtschaftsinformatik				+	+
Internettechnologien				++	+
IT-Sicherheit	++	+	+	+	+
IT-Systeme		++			
Lineare Algebra und Analytische Geometrie	++		+		
Praktisches Studiensemester					++
Praxisbegleitende Lehrveranstaltung					++
Programmieren 1	+		+	+	
Programmieren 2	+		+	+	
Projektarbeit			++	++	++

Projektmanagement			+		++
Rechnerarchitektur		++			
Rechnernetze	+	++			
Seminar					++
Software-Praktikum			++		+
Softwaretechnik 1	+		++		+
Softwaretechnik 2 (HCI)			++		
Theoretische Informatik	++				
Verteilte Softwaresysteme	+	++		+	
Wahrscheinlichkeitsrechnung und Numerik	++		+		

Mit einem Studienabschluss in der Informatik bieten sich den Absolventen heute und auch in Zukunft vielfältige Einsatzmöglichkeiten. In technischen und kaufmännischen Bereichen von Industrieunternehmen, im Handel, bei Banken, Versicherungen und in der öffentlichen Verwaltung werden sie zur Planung, Einsatz und Wartung von Soft- und Hardware benötigt. Ein starker Impuls geht derzeit von diversen Digitalisierungs-Initiativen aus, deren Ziel es ist, sowohl Produkte als auch Dienstleistungen durch Software zu unterstützen.

Es besteht daher ein kontinuierlicher Bedarf an qualifizierten Informatikern in vielen Zweigen der Wirtschaft und im öffentlichen Dienst, der über Jahre hinaus abgedeckt werden muss. Das dynamische wirtschaftliche Wachstum im Bereich der Informationstechnologien eröffnet außerdem engagierten und kreativen Absolventen einen leichten Einstieg in die berufliche Selbständigkeit.

Der Einsatz von DV-Systemen führt zu einem breiten Aufgabenfeld in Firmen jeder Größenordnung. Anwendungen im kaufmännischen und technischen Bereich, lokal oder für weit verteilte Firmenstandorte entwickelt, erfordern firmenspezifisch angepasste DV-Lösungen, die von Informatikern eingeführt und betreut werden.

Darüber hinaus bieten Hard- und Software-Firmen anspruchsvolle Tätigkeiten in den Bereichen Entwicklung, Produktion, Schulung, Kundenbetreuung und Marketing von Software. Um im internationalen Wettbewerb bestehen zu können, benötigen diese Firmen gut ausgebildete Informatiker zur Erstellung und Pflege qualitativ hochwertiger Softwareprodukte.

Aufbau des Studiengangs Informatik

Informatik, Bachelor (B.Sc.)

Hochschule für angewandte Wissenschaften Kempten

gültig mit Studienbeginn WS 2014/15

Semester

7	BA-SEM ⁽⁸⁾	Bachelorarbeit ⁽⁸⁾													WP-Fach ⁽⁶⁾				WP-Fach ⁽⁶⁾				IT-Sicherheit ⁽⁴⁾						
6	Projektarbeit ⁽⁸⁾													Seminar ⁽⁸⁾				WP-Fach ⁽⁶⁾				WP-Fach ⁽⁶⁾							
5	Praktisches Studiensemester ⁽⁷⁾																								Praxisbegleitende Lehrveranstaltung ⁽⁴⁾				
4	Wahrscheinlichkeitsrechnung & Numerik ⁽¹⁾					Internettechnologien ⁽³⁾					Verteilte Softwaresysteme ⁽³⁾					Software Praktikum ⁽³⁾					Rechnernetze ⁽²⁾					Projektmanagement ⁽⁴⁾			
3	Diskrete Mathematik ⁽¹⁾					Datenbanken ⁽³⁾					Softwaretechnik 1 ⁽³⁾					Softwaretechnik 2 ⁽³⁾					Betriebssysteme ⁽³⁾					Compiler ⁽³⁾			
2	Lineare Algebra & Analytische Geometrie ⁽¹⁾					Theoretische Informatik ⁽⁵⁾					Programmieren 2 ⁽³⁾					Algorithmen & Datenstrukturen ⁽³⁾					Rechnerarchitektur ⁽²⁾					AW-Fach ⁽⁴⁾			
1	Analysis ⁽¹⁾					Einführung in die Informatik ⁽³⁾					Programmieren 1 ⁽³⁾										IT Systeme ⁽²⁾					Grundlagen der Wirtschaftsinformatik ⁽⁴⁾			
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30																													

ECTS-Punkte

ECTS-Punkte

Legende:

1	Mathematik	5	Theorie der Informatik
2	Computertechnik	6	Wahlpflichtfach
3	Softwaretechnik	7	Praktikum
4	Angewandte Informatik und interdisziplinäre Fächer	8	Seminar- und Abschlussarbeit

Für den Studiengang Informatik stehen folgende Ansprechpartner zur Verfügung:

Studiengangkoordinator:

Prof. Dr. Elmar Böhler

Studienfachberater:

Prof. Dr. Jochen Staudacher

Beauftragter für das Praxissemester:

Prof. Dr. Patrick Scharpfenecker

Vorsitzender der Prüfungskommission:

Prof. Dr. Ulrich Göhner

2 Begriffserläuterungen

ECTS - European Credit Transfer System

Diese Vereinbarungen zur Anrechnung, Übertragung und Akkumulierung von Studienleistungen basieren auf dem Arbeitspensum, das Studierende durchzuführen haben, um die Ziele des Lernprogramms zu erreichen. Für jede studienbezogene Leistung wird der voraussichtliche durchschnittliche Arbeitsaufwand angesetzt und auf das Studienvolumen angerechnet. Der Arbeitsaufwand umfasst Präsenzzeit und Selbststudium ebenso wie die Zeit für die Prüfungsleistungen, die notwendig sind, um die Ziele des vorher definierten Lernprogramms zu erreichen. Mit dem ECTS können Studienleistungen international angerechnet und übertragen werden.

Arbeitsaufwand (Workload) und Leistungspunkte (ECTS-LP)

Der Arbeitsaufwand der Studierenden wird im ECTS in Credit Points angegeben. Deutsche Übersetzungen für Credit Point sind die Begriffe Leistungspunkt oder ECTS-Punkt. Ein Arbeitsaufwand von 30 Zeitstunden bedeutet einen Leistungspunkt. Der Arbeitsaufwand von Vollzeitstudierenden entspricht 60 Leistungspunkten pro Studienjahr, also 30 Leistungspunkten pro Semester. Das sind 1.800 Stunden pro Jahr oder 45 Wochen/Jahr mit 40 Stunden/Woche.

Der Arbeitsaufwand setzt sich zusammen aus:

- Präsenzzeit
- Zeit für die Vor- und Nachbereitung des Vorlesungsstoffs,
- Zeit für die Vorbereitung von Vorträgen und Präsentationen,
- Zeit für die Erstellung eines Projekts,
- Zeit für die Ausarbeitung einer Studienarbeit,
- Zeit für notwendiges Selbststudium,
- Zeit für die Vorbereitung auf mündliche oder schriftliche Prüfungen.

Die Bachelorstudiengänge mit sieben Semestern bescheinigen erfolgreichen Studierenden 210 ECTS-LP, die dreisemestrigen Masterstudiengänge weitere 90 ECTS-LP. Damit ist die Forderung nach 300 ECTS-LP für ein erfolgreich abgeschlossenes Masterstudium erfüllt.

Semesterwochenstunden und Präsenzzeit

Eine Semesterwochenstunde ist die periodisch wiederkehrende Lehreinheit in einem Modul, in der Regel im Rhythmus von einer oder zwei Wochen. Dabei wird eine Präsenz von 45 Minuten plus Wegzeiten gerechnet, sodass die Vorlesungsstunde als eine Zeitstunde gewertet wird.

Wir rechnen mit einer Vorlesungszeit von 15 Wochen pro Semester, wodurch sich aus der Zahl der Semesterwochenstunden die geforderte Präsenzzeit ("Kontaktzeit") direkt ableitet: 1 SWS entspricht 15 Stunden Präsenzzeit.

Module

Der Studiengang setzt sich aus Modulen zusammen. Ein Modul repräsentiert eine inhaltlich und zeitlich zusammengehörige Lehr- und Lerneinheit. Module werden in der Regel in einem

Semester abgeschlossen.

Modulgruppen sind Zusammenfassungen von Modulen mit einem weiteren inhaltlichen Zusammenhang. In allen Fällen stellt ein Modul oder ein Teilmodul eine Einheit dar, für die innerhalb und am Ende eines Semesters eine Prüfungsleistung erbracht werden kann, für die Leistungspunkte vergeben werden.

Wahlpflichtmodule werden bedarfsorientiert, meist in jährlichem Rhythmus angeboten. Das jeweilige Semester (Sommer- oder Wintersemester) kann der Modulbeschreibung entnommen werden. Grundsätzlich können Wahlpflichtmodule ab einer Untergrenze von 15 angemeldeten Teilnehmern durchgeführt werden. In besonderen Ausnahmefällen, z. B. bei wiederholter Unterschreitung der Mindestteilnehmerzahl oder erstmaligem Angebot einer Veranstaltung, kann die Fakultät von dieser Regelung abweichen. Die Entscheidung treffen der Fakultätsrat und der Dekan der Fakultät. Wahlpflichtmodule, die gleichzeitig in einem anderen Studiengang als Pflichtmodul gehalten werden, können auch stattfinden, wenn weniger als 15 Anmeldungen vorliegen.

Die Lehrveranstaltungen werden derzeit in deutscher Sprache gehalten.

Studienbegleitende Prüfungen und Studienfortschritt

Sämtliche Prüfungen erfolgen über das gesamte Studium verteilt studienbegleitend und stehen in direktem Bezug zur Lehrveranstaltung. Prüfungsbestandteile können je nach Lehrveranstaltung begleitend oder nach Abschluss des Moduls stattfinden, beispielsweise als Referat, Klausurarbeit, mündliche Prüfung, Hausarbeit mit Kolloquium, Entwurf mit Kolloquium, Laborbericht, Exkursionsbericht oder einer Kombination. In den Beschreibungen der einzelnen Module wird im Modulhandbuch die jeweilige Prüfungsform festgelegt. Es gelten allgemeine Studienfortschrittsberechtigungen, die die jeweils gültige Studien- und Prüfungsordnung regelt.

3 Modulbeschreibungen

IFB1101 Einführung in die Informatik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Rieck
Dozent(en):	Prof. Dr. Stefan Rieck
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage,

- die Informationsdarstellung im Rechner zu beschreiben
- einfache Algorithmen und Datenstrukturen wiederzugeben
- die Vorgehensweise beim Übersetzen von Programmen zu erläutern
- einfache Automaten und Sprachen zu definieren
- die wichtigsten Schritte bei der Software-Entwicklung zu erläutern
- die wirtschaftlichen und gesellschaftlichen Auswirkungen der Informatik zu diskutieren

Lehrinhalte:

- Grundbegriffe der Informatik; Nachricht und Information, Codierung; Zahlensysteme
- Einführung in Algorithmen und Datenstrukturen
- Grundlagen der Automatentheorie und formaler Sprachen
- Grundlagen der Softwareentwicklung
- Wirtschaftliche und gesellschaftliche Verantwortung der Informatik

Literatur:

- H.P. Gumm, M. Sommer: Einführung in die Informatik, 10. Auflage, Oldenburg, 2012

IFB1102 Analysis

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Christian Blick
Dozent(en):	Prof. Dr. Christian Blick
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht/Übung 30 Stunden Selbststudium - Betreute Studierzeit 60 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme am Modul Analysis können die Studierenden

- Grundbegriffe der Analysis wiedergeben
- wesentliche Berechnungsmethoden der Analysis problemabhängig auswählen
- einfache Berechnungen mit Methoden der Analysis durchführen
- einfache Beweise analysieren und einfache Beweismethoden anwenden

Lehrinhalte:

- Aussagenlogik, Grundbegriffe der Prädikatenlogik
- Natürliche Zahlen und Vollständige Induktion
- Rationale Zahlen, Funktionen, Reelle Zahlen und Körperaxiome
- Folgen und Reihen, Grenzwerte
- Stetigkeit und stetige Funktionen
- Differential- und Integralrechnung
- Taylor-Entwicklung, Potenzreihen
- Differentialgleichungen: Beispiele

Literatur:

- Brill, M.: Mathematik für Informatiker, Hanser Verlag, 2. Auflage, 2005
- Hachenberger, D.: Mathematik für Informatiker, Pearson Studium, 2. Auflage, 2008
- Hartmann, P.: Mathematik für Informatiker, Springer Vieweg Verlag, 2019
- Stingl, P.: Einstieg in die Mathematik für Fachhochschulen, Hanser Verlag, 2013

-
- Stingl, P.: Mathematik für Fachhochschulen, Hanser Verlag, 2007

IFB1103 Programmieren 1

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht 4 SWS betreutes Praktikum in kleinen Gruppen
Leistungspunkte:	10
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht 60 Stunden Präsenzzeit Praktikum 180 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 120 Minuten am Ende des Semesters Teilnahmepflicht im Praktikum, Leistungsnachweise im Praktikum Leistungsnachweise sind Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden können kleine Probleme der realen Welt mit prozeduralen oder objektorientierten Programmen lösen.
- Die Studierenden sind in der Lage, eine IDE (Integrated Development Environment) zum Programmieren, Übersetzen und zur Fehlersuche zu benutzen.

Lehrinhalte:

- Einführung in C und C++
- Grundlegende Datentypen
- Programmablaufkontrollstrukturen (Selektion, Schleifen)
- Verständnis des Übersetzungsvorgangs (Präprozessor, Compiler, Linker)
- Funktionen und Möglichkeiten der Parameterübergabe
- Strukturierte Datentypen (structs), Felder (Arrays) und Zeichenketten (Strings)
- Statische vs. dynamische Speicherverwaltung mittels Pointern
- Verwendung eigener und fremder Bibliotheken
- Statisches vs. dynamisches Linken gegen Bibliotheken
- Lesen und Schreiben von Daten aus/in Dateien
- Grundkonzepte der objektorientierten Programmierung (Klassen, Datenkapselung, Vererbung)

- Arbeiten mit einer IDE (Visual Studio)

Literatur:

Einführung in die Programmiersprache C:

- Joachim Goll, Manfred Dausmann. C als erste Programmiersprache. Mit den Konzepten von C11. SpringerVieweg Verlag. 8. Auflage, 2014.

Einführung in die Objektorientierte Programmierung:

- Stefan Rieck. OOP für Ingenieure. Beispiele in C++. VDE-Verlag, 2002.

IFB1104 Lineare Algebra und Analytische Geometrie

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Christian Blick
Dozent(en):	Prof. Dr. Christian Blick
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.), Studiengang Gesundheits- und Pflegeinformatik (B.Sc.), , Studiengang Medizininformatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung in kleinen Gruppen (14tägig 90 Minuten)
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit Unterricht 15 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung können die Studierenden

- grundlegende Begriffe der Linearen Algebra definieren
- einfache geometrische Probleme in der Ebene und im Raum visualisieren, mathematisch beschreiben und lösen
- sicher mit Matrizen und Vektoren arbeiten
- die Lösbarkeit eines linearen Gleichungssystems beurteilen und die Struktur von dessen Lösungsmenge beschreiben
- mathematische Sachverhalte klar formal kommunizieren
- Methoden der Linearen Algebra auf einfache fachbezogene Fragestellungen anwenden

Lehrinhalte:

- Grundbegriffe der Mengenlehre, Relationen und Abbildungen
- Vektorräume, lineare und affine Unterräume
- Arbeiten mit Vektoren in der Ebene und im Raum
- Skalarprodukt, Orthogonalität, Normen, Längen- und Winkelmessung
- Lineare Abbildungen, Matrizenrechnung
- Matrizen mit besonderen Eigenschaften, Drehungen, Spiegelungen

- Lineare Gleichungssysteme und deren Lösungsmengen
- Lösungsverfahren für Lineare Gleichungssysteme
- Eigenwerte und Eigenvektoren

Literatur:

- M. Plaue, M. Scherfner: Mathematik für das Bachelorstudium I: Grundlagen und Grundzüge der linearen Algebra und Analysis, Spektrum Akademischer Verlag, 2. Auflage, 2019
- G. Gramlich: Lineare Algebra: Eine Einführung, Hanser Fachbuchverlag, 4. Auflage, 2014
- D. Hachenberger: Mathematik für Informatiker, Pearson Studium, 2. Auflage, 2008
- J. Schwarze: Mathematik für Wirtschaftswissenschaftler. Band 1: Grundlagen, Nwb Verlag, 14. Auflage, 2015

IFB1105 Programmieren 2

Allgemeines

Programmieren 2 setzt grundlegende Kenntnisse in der Programmiersprache C aus der Erstsemestervorlesung Programmieren 1 voraus. In dieser Veranstaltung werden darauf aufbauend Programmierkonzepte anhand der Programmiersprache Modern C++ vermittelt. Dazu gehören erste Programmiermuster, einfache Algorithmen, erweiterte Konzepte der OOP (Klassen, Templates, Exception Handling, Multithreading) sowie zusätzliche Programmierkonzepte wie generische Programmierung und Operatorüberladung in C++ vermittelt und praktisch umgesetzt.

Modulverantwortliche(r):	Prof. Dr. Bernd Lüdemann-Ravit
Dozent(en):	Prof. Dr. Bernd Lüdemann-Ravit
Modultyp:	Pflichtmodul
Voraussetzungen:	Programmieren 1
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS betreutes Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	kein programmierbarer Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden sind in der Lage, bei der Lösung von Programmieraufgaben strukturiert vorzugehen.
- Die Studierenden können kleine Probleme der realen Welt (gegebenenfalls mit Hilfe von Standard-Bibliotheken und Frameworks) lösen.

Lehrinhalte:

- Grundlegende Schritte des Software-Entwicklungsprozesses (Analyse, Design, Implementierung, Test)
- Basis-Einführung in Entwurfsmuster
- Einsatz von Standard-Bibliotheken am Beispiel der C++ Standard Library
- Grundlagen der Programmierung in Modern C++
- Einsatz von Frameworks
- Festigung des Wissens zur OOP
- Templates in C++
- Operator Überladung in C++

Literatur:

- Gamma, Helm, Johnson, Vlissides: Entwurfsmuster. Elemente wiederverwendbarer Software, Addison-Wesley, München, 2014
- Stroustrup, Björn: A Tour of C++. Addison-Wesley, 2022
- Meyers, Scott: Effective Modern C++: 42 Specific Ways to Improve Your Use of C++11 and C++14 O'reilly, 2014
- Meyers, Scott: Effective C++: 55 Specific Ways to Improve Your Programs and Designs, Addison-Wesley, 2005
- Meyers, Scott: More Effective C++: 35 New Ways to Improve Your Programs and Designs, 1995
- <http://www.cplusplus.com/>

IFB1106 Algorithmen und Datenstrukturen

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Ulrich Göhner
Dozent(en):	Prof. Dr. Ulrich Göhner
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzung
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden haben einen Überblick über wichtige Algorithmen und Datenstrukturen in der Informatik. Die Studierenden sind in der Lage, die Aufwände der Algorithmen zu quantifizieren und in Komplexitätsmaße zu fassen. Sie können eine gegebene Problemstellung in einen Algorithmus fassen und diesen in ein effizient laufendes Programm überführen.

Lehrinhalte:

- Definition und Darstellung Algorithmus
- Komplexität und O-Notation
- Greedy-Algorithmen, Rekursion, Divide and Conquer-Algorithmen
- Definition Datenstruktur / ADT
- verkettete Liste, Stapel und Schlangen
- Tabelle mit Zugriffsoperationen und Implementierungen
- elementare Sortierverfahren (Insertion-, Selection-, ExchangeSort)
- schnelle Sortierverfahren (Quicksort, Mergesort, Heapsort, Shellsort, Combsort)
- Baumstrukturen (Begriffsbildung, Suchbaum, Heap, Treap, ausgeglichene Bäume)
- Hashfunktion
- Graphen (Definition, Darstellung, Implementierung, Breiten/Tiefensuche, reflexive transitive Hülle, Kürzeste Wege)

Literatur:

- Sedgewick, R.; Wayne, K.: "Algorithmen", 4. Auflage, Pearson, 2014.
- Saake, G.; Sattler, K.-U.: "Algorithmen und Datenstrukturen", 5. Auflage, dpunkt.verlag, Heidelberg, 2014.
- Reeß, H.; Viebeck, G.: "Datenstrukturen und Algorithmen (C++)", Carl Hanser Verlag, 2002.
- Heun, V.: "Grundlegende Algorithmen (C)", Vieweg Verlag, 2. Auflage, 2003.
- Dietzfelbinger, M.; Mehlhorn, K., Sanders, P.: "Algorithmen und Datenstrukturen", Springer, Berlin Heidelberg, 2014.
- Edmonds: How to think about Algorithms, Cambridge, 2008

IFB1107 Theoretische Informatik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Ulrich Göhner
Dozent(en):	Prof. Dr. Ulrich Göhner / Prof. Nikolaus Steger
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzung
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden kennen die wichtigsten Grundlagen der theoretischen Informatik. Sie beherrschen die Grundbegriffe der Automatentheorie, der formalen Sprachen, der Berechenbarkeit und der Entscheidbarkeit.

Lehrinhalte:

- Formale Sprachen und Grammatik
- Chomsky-Hierarchie
- Reguläre Sprachen und reguläre Ausdrücke
- Automatentheorie (NEA, DEA)
- Thompson-Algorithmus
- Turing-Berechenbarkeit
- Entscheidbarkeit
- Komplexitätstheorie, NP-Vollständigkeit

Literatur:

- Hopcroft, J.E., Motwani, R., Ullman, J.D.: "Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie", Addison-Wesley
- Aho, A.V., Sethi, R., Ullmann, J. D.: "Compilerbau", Band 1 und 2, Addison-Wesley 1988
- Aho, A.V., Lam, M.S., Sethi, R., Ullmann, J. D.: "Compiler", Addison Wesley
- Vossen, G., Witt, K.-U.: "Grundkurs Theoretische Informatik", Vieweg

- Asteroth, A., Baier, Ch.: "Theoretische Informatik", Pearson
- Hromkovic, J.: "Theoretische Informatik", Springer Vieweg, 2015.
- Wagenknecht, Ch.; Hielscher, M.: "Formale Sprachen, abstrakte Automaten und Compiler", Springer Vieweg, 2014.
- Böckenhauer, H.-J., Hromkovic, J.: "Formale Sprachen, Endliche Automaten, lexikalische und syntaktische Analyse", Springer Vieweg Wiesbaden, 2013.

IFB1108 IT-Systeme

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Rafael Mayoral Malmström
Dozent(en):	Prof. Dr. Rafael Mayoral Malmström
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung in Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit Unterricht 15 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage,

- den grundlegenden Aufbau, die Komponenten und die Funktionsweise von programmierbaren Systemen zu erläutern
- einfache Assembler-Programme zu implementieren.

Lehrinhalte:

Die Lehrveranstaltung vermittelt Grundlagen folgender Themen:

- Boole'sche Algebra, Digitale Logik, Digitale Grundsaltungen,
- Rechenwerk, Steuerwerk,
- Speicher, Peripherie,
- Mikroprogrammierung, Assemblerprogrammierung.

Literatur:

- A. Tanenbaum: Rechnerarchitektur: Von der digitalen Logik zum Parallelrechner, Pearson Studium - IT, 2014
- D. W. Hoffmann: Grundlagen der Technischen Informatik, Hanser, 5. Auflage, 2016
- H.-P. Gumm, M. Sommer: Einführung in die Informatik, Oldenbourg Verlag, 2012

IFB1109 Rechnerarchitektur

Allgemeines

Rechnerarchitektur ist ein Teilgebiet der Technischen Informatik, das sich mit dem Design von Rechnern und speziell mit deren Organisation sowie deren externem und internem Aufbau beschäftigt.

Modulverantwortliche(r):	Prof. Dr. Rafael Mayoral Malmström
Dozent(en):	Prof. Dr. Rafael Mayoral Malmström
Modultyp:	Pflichtmodul
Voraussetzungen:	IT-Systeme
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung in Gruppen
Leistungspunkte:	6
Arbeitsaufwand:	45 Stunden Präsenzzeit Unterricht 15 Stunden Präsenzzeit Übung 120 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie, nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Nach dem Besuch dieser Lehrveranstaltung, sind die Studierenden in der Lage:

- Leistung und Zuverlässigkeit von Rechnersystemen zu definieren, zu berechnen und zu bewerten.
- das Konzept von Benchmarks zu verstehen, diese einzusetzen und Rechnersysteme anhand ihrer Ergebnisse zu bewerten.
- die Prinzipien verschiedenartiger Rechnerarchitekturen zu verstehen und zu beschreiben.
- grundlegende Techniken der modernen Mikroarchitekturen von Prozessoren zu benennen, ihre Funktionsprinzipien zu beschreiben und verschiedene Varianten anhand von Simulationen und Berechnungen zu bewerten und zu vergleichen.
- die Wichtigkeit des Speichersubsystems eines Rechners für die Gesamtleistung zu verstehen und zu begründen.
- unterschiedliche Cache-Realisierungen zu verstehen und zu bewerten.
- verschiedene Techniken zur Parallelisierung zu benennen, deren Funktionsprinzip zu beschreiben und zu bewerten.

Lehrinhalte:

Die Veranstaltung vermittelt Grundlagen der folgenden Themen:

- Grundlegende Konzepte der Rechnerarchitektur und -organisation
- Unterschiedliche Prozessor-Architekturen
- Leistung und Leistungsbewertung
- Befehlssatzarchitektur: Speichermodell, Register, Befehlssatz
- Mikroarchitektur: Umsetzung der Befehlssatzarchitektur, Datenpfad, Pipelining, Hazards, Sprungvorhersage, Superskalarität, Out-of-Order-Ausführung
- Der Speichersubsystem: Cache-Speicher, Schreibstrategien, Cache-Leistung
- Parallelisierung: On-Chip Multithreading, Multiprozessoren und Multicomputer
- Zuverlässigkeit

Literatur:

- David A. Patterson, John L. Hennessy, Computer Organization and Design: MIPS Edition, Morgan Kaufmann, 5th ed., 2014
- David A. Patterson, John L. Hennessy, Computer Organization and Design: RISC-V Edition, Morgan Kaufmann, 5th ed., 2018
- Andrew S. Tanenbaum, Todd Austin: Rechnerarchitektur, Pearson, 6. Aufl., 2014
- Andrew S. Tanenbaum, Herbert Bos: Moderne Betriebssysteme, Pearson, 4. Aufl., 2016
- Axel Böttcher, Rechneraufbau und Rechnerarchitektur, Springer-Verlag Berlin Heidelberg 2006

IFB1110 Grundlagen der Wirtschaftsinformatik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Michael Lenke
Dozent(en):	Prof. Dr. Michael Lenke
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Erfolgreiche Teilnahme am Praktikum gilt als Zulassungsvoraussetzung für die 90 minütige schriftliche Prüfung, die am Ende des Semesters erfolgt.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Absolventen dieses Moduls können:

- Information als kritischen Erfolgsfaktor für Unternehmen erkennen;
- unterschiedliche Klassifikationen für IT-Systeme gegenüberstellen und anwenden;
- den typischen Lebenszyklus eines IT-Systems beschreiben und die dabei benötigten Kompetenzen benennen;
- zentrale Begriffe der Wirtschaftsinformatik nennen, einordnen und anwenden;
- Integrationsansätze moderner IT-Systeme diskutieren;
- typische E-Business Anwendungen als Bestandteil moderner IT-Infrastrukturen erkennen und einordnen;
- im Rollenspiel des Praktikums im Rahmen eines IT-System-Einführungsprojektes Aspekte Ihres späteren Einsatzgebietes als Wirtschaftsinformatiker erfahren.

Lehrinhalte:

- Klassifikationen von IT-Systemen
- Life Cycle von IT-Systemen
- Zentrale Begriffe der Wirtschaftsinformatik
- Ausgewählte Themen der Informatik, Wirtschaftsinformatik und Betriebswirtschaft:
 - Architektur und Technologien moderner IKS.
 - E-Business Anwendungen
 - Menschen und Unternehmen im Kontext "Wirtschaft - Wirtschaftsordnung - Geldwirtschaft - Geldkreislauf - Geld"

Literatur:

- Bernd W. Wirtz: Electronic Business, 4. Auflage, 2013, Springer Gabler
- Franz Lehner, S. Wildner, M. Scholz: Wirtschaftsinformatik - Eine Einführung, 2008, Carl Hanser Verlag
- Hansen, Mendling, Neumann: Wirtschaftsinformatik, 2019; De Gruyter Oldenburg
- R. Thome: Grundzüge der Wirtschaftsinformatik. Pearson Studium 2006

IFB1111 Allgemeinwissenschaftliches Wahlpflichtfach

Allgemeines

Modulverantwortliche(r):	Professoren und Lehrbeauftragte, verantwortlich: AW Koordinator
Dozent(en):	Professoren und Lehrbeauftragte, verantwortlich: AW Koordinator
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	abhängig von der Belegung
Leistungspunkte:	4
Arbeitsaufwand:	Abhängig von der Belegung
Leistungsnachweis und Prüfung:	siehe Modulhandbuch Allgemeinwissenschaftliche Module
Zur Prüfung zugelassene Hilfsmittel:	siehe Modulhandbuch Allgemeinwissenschaftliche Module

Lernergebnisse und Inhalte

Lernergebnisse:

Die Fächer (Ziele) werden im Einvernehmen mit dem AW-Beirat beschlossen und können im Katalog der Allgemeinwissenschaftlichen Module nachgelesen werden.

Lehrinhalte:

Die Fächer können aus folgenden Bereichen ausgewählt werden:

- Historisch-politischer Bereich
- Philosophisch-ethischer Bereich
- Psychologisch-, pädagogisch-, soziologischer Bereich
- Technisch-naturwissenschaftlicher Bereich
- Wirtschafts- und Rechtswissenschaften
- Kommunikation und Rhetorik
- Arbeits- und Kreativitätsmethoden
- Sprachlicher Bereich

Literatur:

Literaturempfehlungen in Absprache mit den zuständigen Dozenten

IFB1112 Diskrete Mathematik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Christian Blick
Dozent(en):	Prof. Dr. Christian Blick
Modultyp:	Pflichtmodul
Voraussetzungen:	Lineare Algebra
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Vorlesung/Übung 30 Stunden Selbststudium - Betreute Studierzeit 60 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme am Modul Diskrete Mathematik können die Studierenden

- wesentliche Begriffe der Diskreten Mathematik wiedergeben
- wesentliche Berechnungsmethoden der Diskrete Mathematik problemabhängig auswählen
- Berechnungen mit Methoden der Diskrete Mathematik durchführen

Lehrinhalte:

- Grundlegende Begriffe der Diskreten Mathematik
- Abzählmethoden
- Graphen, Bäume, aufspannende Bäume, kürzeste Wege, Euler- und Hamilton-Kreise
- Algorithmen von Kruskal und Dijkstra, MST-Heuristik
- Determinanten
- Ringe: Matrizenring, Polynomring
- Körper: Komplexe Zahlen, Endliche Körper, modulare Arithmetik,
- Einführung in die Lineare Optimierung

Literatur:

- Aigner, M.: Diskrete Mathematik, Vieweg Verlag, 2007
- Aigner, M.: Zahlentheorie : eine Einführung mit Übungen, Hinweisen und Lösungen, Vieweg-Teubner, 2012
- Brill, M.: Mathematik für Informatiker, Hanser Verlag, 2. Auflage, 2005

-
- Hachenberger, D.: Mathematik für Informatiker, Pearson Studium, 2. Auflage, 2008
 - Hartmann, P.: Mathematik für Informatiker, Springer Vieweg Verlag, 2019
 - Stingl, P.: Mathematik für Fachhochschulen, Hanser Verlag, 2007
 - Stingl, P.: Operations Research : Linearoptimierung, Hanser Verlag, 2002

IFB1113 Datenbanken

Allgemeines

Die Studierenden haben einen Überblick über den Einsatz von Datenbanken und Informationssystemen. Sie kennen die Grundlagen und die praktische Anwendung von Datenmodellierung und Datenbankentwurf. Die Studierenden können Datenbanken mit SQL erstellen, die Daten manipulieren und komplexe Abfragen durchführen. Sie sind in der Lage Ausführungspläne zu analysieren und Optimierungen am physischen Entwurf vorzunehmen. Außerdem können sie transaktionsorientierte Anwendungen erstellen.

Modulverantwortliche(r):	Prof. Nikolaus Steger
Dozent(en):	Prof. Nikolaus Steger
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage

- Datenmodelle aus gegebenen Anforderungen zu erstellen
- Einen Logischen Datenbankentwurf und einen Physischen Datenbankentwurf aus einem gegebenen Datenmodell zu entwerfen und Datenbank mit SQL DDL zu erstellen
- Daten in Datenbanken mit SQL zu manipulieren
- Komplexe Abfragen in Relationaler Algebra und in SQL auf einem gegebenen Datenbankschema zu erstellen
- SQL-Abfragen in Relationale Algebra zu übersetzen und mittels Heuristischer Optimierung zu optimieren
- Zugriffspläne zu Abfragen zu analysieren und daraus physische Optimierungen abzuleiten und umzusetzen
- Ein Datenbankschema mit gegebenen Funktionalen Abhängigkeiten bis zur Boyce-Codd-Normalform zu normalisieren

- Transaktionsorientierte Anwendungen mit SQL zu implementieren

Lehrinhalte:

- Grundlagen Datenbanken und Informationssysteme
- Entity Relationship-Datenmodelle und Datenmodellierung mit UML
- Das Relationale Datenmodell
- Relationale Algebra
- Datenbankentwurf
- SQL (DDL, DML und DCL)
- Datenbankzugriff aus Programmiersprachen
- Implementierung der Relationalen Algebra
- Query-Übersetzung und Optimierung
- Relationale Entwurfstheorie, Normalformen
- Das ACID-Prinzip

Literatur:

- Kemper, Eickler: Datenbanksysteme: Eine Einführung, 10. Auflage 2015, Oldenburg,
- Elmasri, Navathe: Fundamentals of Database Systems, Pearson, 7th Edition 2016, Pearson
- Garcia-Molina, Ullman, Widom: Database Systems The Complete Book, Second Edition 2013, Pearson

IFB1114 Softwaretechnik 1 / Software Engineering

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Georg Hagel
Dozent(en):	Prof. Dr. Georg Hagel
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweise in der Übung, schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzungen.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage,

- verschiedene Vorgehensmodelle mit ihren Stärken und Schwächen zu beschreiben.
- die UML in ihrer aktuellen Version zur Beschreibung von Ergebnissen in Analyse, Architektur und Design anzuwenden.
- alle Phasen der Softwareerstellung (Requirements Engineering, Analyse, Architektur und Design, Implementierung und Qualitätssicherung) zu beschreiben.
- Bekannte Muster in Analyse und Entwurf anzuwenden.
- Testfallermittlung und Metriken auf gegebene Problemstellungen anzuwenden.

Lehrinhalte:

- Vorgehensmodelle
- Modellierung mit Strukturdiagrammen
- Modellierung mit Verhaltensdiagrammen
- Modellierung mit Architekturdiagrammen
- Modellierung mit Interaktionsdiagrammen
- Requirements Engineering
- Analyse und Analysemuster
- Architekturbeschreibung
- Design-Beschreibung und Design-Muster

- Qualitätssicherung
- Ethik in der Informatik

Literatur:

- Sommerville, Ian: Software Engineering, Pearson Studium, 10. Auflage (2018)
- Farley, David: Modern Software Engineering, Addison-Wesley Professional (2022)
- Oestereich, Bernd: Analyse und Design mit der UML 2.5, Oldenbourg Verlag, 11. Auflage (2013)
- Kecher, Christoph: UML 2.5 –Das umfassende Handbuch, Rheinwerk Verlag, 7. Auflage (2021)

IFB1115 Betriebssysteme

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Arnulf Deinzer
Dozent(en):	Prof. Dr. Arnulf Deinzer
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Pflicht zur aktiven Teilnahme in Übungen/Praktikum, Schriftliche Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie, nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden haben fundierte Kenntnisse aus Aufgaben und Realisierungen von Betriebssystemen. Sie sind sicher im Umgang mit Linux auf Nutzerniveau und sind nach erfolgreicher Beendigung der Veranstaltung in der Lage, in einem experimentellen Betriebssystem einfache Aufgaben aus Scheduling und Synchronisation von Prozessen umzusetzen.

Im Einzelnen sind die Studierenden nach erfolgreicher Beendigung der Veranstaltung in der Lage

- die gebräuchlichsten Verfahren für die Organisation von Sekundärspeichern darzustellen und die entsprechenden Verwaltungsaufwände zu berechnen
- verschiedene Schedulingstrategien darzustellen und unter Verwendung von Standardmetriken zu vergleichen
- die gebräuchlichsten Verfahren für die Organisation von Primärspeichern darzustellen und die entsprechenden Verwaltungsaufwände zu berechnen
- sicher mit grundlegenden Synchronisationsprimitives wie Semaphoren umzugehen, d.h. ein gegebenes Synchronisationsproblem mit Semaphoren zu lösen bzw. problematische Implementierungen zu verbessern
- Strategien zur Deadlockerkennung und -vermeidung darzustellen und in Vor- und Nachteilen zu vergleichen

Lehrinhalte:

- Einführung und Überblick
- Dateisysteme, IO-Devices, Primär-, Sekundär- und Tertiärspeicher
- Prozess- und Prozessorverwaltung
- Primärspeicherverwaltung
- Prozesskommunikation
- Sicherheit
- Kommandosprachen

Literatur:

- Tanenbaum, Andrew S.: Moderne Betriebssysteme, Prentice-Hall 2002
- Stallings, William: Betriebssysteme, 4. Auflage, 2003 Pearson
- Herrtwich R.G., Hommel G.: Kooperation und Konkurrenz, Springer 1989

IFB1116 Internettechnologien

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Katja Bochtler
Dozent(en):	Prof. Dr. Katja Bochtler
Modultyp:	Pflichtmodul
Voraussetzungen:	Softwareentwicklung und Programmieren
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Praktikum in kleinen Projektteams
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	In diesem Modul wird die Prüfungsstudienarbeit im laufenden Semester angefertigt bzw. durchgeführt. Die Prüfungsstudienarbeit kann dabei aus schriftlichen Ausarbeitungen, Präsentationen, Arbeiten am PC oder der Bearbeitung von Aufgabenstellungen im Rahmen einer Projektarbeit und einem Abschlussbericht bestehen. Der Arbeitsumfang beträgt 40 - 60 Stunden.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Absolventen dieses Moduls können:

- Das Internet als Infrastruktur für Web-Technologien verstehen.
- Web Technologien bei der Implementierung moderner IT-Systeme richtig einordnen und anwenden.
- Websites mittels HTML, CSS und JavaScript erstellen.
- Web Applikationen mittels REST- und HTML5-Technologien programmieren.
- XML-Technologien verstehen und am Beispiel des Single Source Publishing (SSP) anwenden.
- Im Praktikum anhand von Praxisbeispielen das Erlernte einüben und weiterführende Techniken kennenlernen.
- Umgang und Einsatzgebiete mit aktuelle Webtechnologien wie react.js, angular.io, vue.js und TypeScript.

Lehrinhalte:

- Grundlagen von Internettechnologien
- Programmierung von Webseiten (HTML, CSS, JavaScript)
- Programmierung von WebApplikationen (react, angular, TypeScript, vue)

- Programmierung mit XML / XSL
- Webservices (REST, SOAP)
- Datentransfer mit AJAX und Node Injection

Literatur:

- <http://www.selfhtml.org/>
- Balzert; Basiswissen Web-Programmierung, 2017; Springer
- Heiko Wöhr; Web-Technologien, 2004; dpunkt.verlag Heidelberg
- Head First: HTML5 Programming, 2011, O'Reilly
- <https://www.typescriptlang.org/>
- <https://reactjs.org/>
- <https://angular.io/>
- <https://vuejs.org/>

IFB1117 Human Computer Interaction / Softwaretechnik 2

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Bernd Dreier
Dozent(en):	Prof. Dr. Bernd Dreier
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.), Studiengang Medizininformatik (B.Sc.)
Angebot und Dauer:	Wintersemester und Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS betreutes Praktikum in kleinen Gruppen mit Erstellung einer Studienarbeit
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht im Praktikum, der endnotenbildende Leistungsnachweis besteht aus einer benoteten Studienarbeit (30-40 Seiten).
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden verstehen die Bedeutung der Human Computer Interaction und ihre Ziele.
- Die Studierenden kennen aktuelle Methoden zur benutzerzentrierten Softwareentwicklung als Design Thinking-Prozess und können diese anwenden.
- Die Studierenden verstehen die relevanten Grundlagen der menschlichen Physiologie und Psychologie und können diese bei der Gestaltung von Benutzeroberflächen anwenden.
- Die Studierenden kennen die relevanten objektiven Kriterien (Normen und Richtlinien) und können diese anwenden.

Lehrinhalte:

- Begriffsdefinition und Einführung von Human Computer Interaction, Interaktionsdesign, Usability (Engineering)
- Einführung in Design Thinking und UI-/UX-Design als strukturierter Prozess
- Einführung eines Prozesses zur Nutzer- und Kontextanalyse (User Empathy Maps, User Needs und Szenarien)
- Relevante Teile der Physiologie und Psychologie des Menschen sowie abgeleitete Modelle und Verfahren
- Objektive Kriterien zur Gestaltung und Beurteilung von User Interfaces, Richtlinien

und Normen, insbesondere DIN EN ISO 9241

Literatur:

- David Benyon, Designing Interactive Systems, 4th edition, 2019, Pearson
- Richter, Flückiger, Usability Engineering kompakt, 3. Auflage, 2013
- Norm DIN EN ISO 9241 in der aktuellen Fassung
- Andreas Butz, Antonio Krüger, Sarah Theres Völkel, Mensch-Maschine-Interaktion, De Gruyter Studium, 3. Auflage, 2022

IFB1118 Compiler

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Ulrich Göhner
Dozent(en):	Prof. Dr. Ulrich Göhner
Modultyp:	Pflichtmodul
Voraussetzungen:	Theoretische Informatik, insbesondere Automatentheorie und formale Sprachen
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden kennen den Aufbau eines Compilers und die verwendeten Methoden beim Compilerbau. Sie beherrschen die Grundbegriffe der Scanner- und Parsertheorie und sind in der Lage, Compiler für einfache Sprachen selbst zu entwickeln. Sie können mit Scanner- und Parsergeneratoren umgehen und kennen die dabei angewandten Techniken.

Lehrinhalte:

- Compiler/Interpreter
- T-Diagramm, Bootstrapping
- Lexikalische Analyse (Adhoc-Scanner, Scanner-Generator, ex)
- Syntaktische Analyse (Top-Down, Bottom-Up, Rekursiver Abstieg, LL-Parsing)
- First- und Follow-Mengen
- Kellerautomat
- LR-Parsing
- Parsergenerator yacc
- Fehlerbehandlung
- Semantische Analyse

- Attributierter Syntaxbaum
- Codegenerierung und Codeoptimierung

Literatur:

- Aho, A.V., Sethi, R., Ullmann, J. D.: "Compilerbau" Band 1 und 2, Addison-Wesley, 1988
- Aho, A.V., Lam, M.S., Sethi, R., Ullmann, J. D.: "Compiler", Addison-Wesley.
- Levine, J. R., Mason, T., Brown, D.: "lex and yacc", O'Reilly and Associates, 1995.
- Wilhelm, R., Seidl, H., Hack, S.: "Übersetzerbau", Springer Vieweg Berlin Heidelberg, 2012.
- Wagenknecht, Ch.; Hielscher, M.: "Formale Sprachenm abstrakte Automaten und Compiler", Springer Vieweg, 2014.
- Böckenhauer, H.-J., Hromkovic, J.: "Formale Sprachen, Endliche Automaten, lexikalische und syntaktische Analyse", Springer Vieweg Wiesbaden, 2013.

IFB1119 Rechnernetze

Allgemeines

OSI-7-Schichtenmodell in Theorie und TCP/IP in Praxis.

Wie funktionieren Rechnernetze - das kommunikative Rückgrat unserer Gesellschaft!

Modulverantwortliche(r):	Prof. Dr. Arnulf Deinzer
Dozent(en):	Prof. Dr. Arnulf Deinzer
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übungen/Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Pflicht zur aktiven Teilnahme in Übungen/Praktikum, Schriftliche Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden kennen das OSI-7-Schichten-Modell und - zum Vergleich - das TCP/IP-Modell. Sie kennen die grundlegenden Aufgaben in den Schichten 1-4 einschließlich der entsprechenden Realisierungen im TCP/IP-Stack. Die Studierende sollen wenigstens für eine Applikation auf L7 exemplarisch zeigen können, wie die Dienste der niedrigeren Schichten genutzt werden.

Im Einzelnen sind die Studierenden nach erfolgreicher Beendigung der Veranstaltung in der Lage

- ein Schichtenmodell auf eine skizzierte Kommunikationsaufgabe anzuwenden
- die Aufgaben der einzelnen Schichten sowohl im OSI- als auch im TCP/IP-Modell zu nennen und einander gegenüberzustellen
- Fehlererkennungs- und -korrekturverfahren der Sicherungsschicht zu beschreiben und deren Leistungsfähigkeiten und Grenzen abzuschätzen
- Mehrfachzugriffsverfahren anschaulich zu beschreiben
- Die gebräuchlichsten Routingprotokolle zu beschreiben und ihre Realisierung in einfachen Beispielnetzen umzusetzen
- Vor- und Nachteile von verbindungsorientierten und verbindungslosen Transportprotokollen zu nennen und einander gegenüberzustellen

Lehrinhalte:

- Einführung
- Modelle (OSI-Referenzmodell, TCP/IP-, Normierungsgremien)
- Bitübertragungsschicht
- Sicherungsschicht (incl. MAC, LLC), LAN
- Vermittlungsschicht
- Transportschicht
- Verarbeitungsschicht

Literatur:

- Tanenbaum, Andrew S.: Computernetzwerke (4. Auflage), Pearson 2003
- Kurose, James F., Ross, Keith W.: Computernetze, Pearson 2002

IFB1120 Verteilte Softwaresysteme

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Frenz
Dozent(en):	Prof. Dr. Stefan Frenz
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.), Studiengang Medizininformatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übungen/Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	schriftl. Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden verstehen die grundlegenden Herausforderungen an und Möglichkeiten von softwareseitig verteilten Systemen. Sie lernen das OSI-7-Modell, Client-/Server- und Peer-To-Peer-Architekturen sowie die erforderlichen theoretischen Hintergründe kennen. Die Studierenden implementieren verteilte Anwendungen über UDP, TCP und RMI. Die Studierenden evaluieren und bewerten Konsistenzmodelle für verteilte Anwendungen.

Lehrinhalte:

- Kommunikationsgrundlagen
- Architektur verteilter Systeme
- Implementierung verteilter Anwendungen
- Konsistenzmodelle verteilter Anwendungen
- Sicherheitsaspekte verteilter Anwendungen
- Beispiele verteilter Anwendungen

Literatur:

- Tanenbaum, Andrew S.; van Steen, Marten: Verteilte Systeme (2. Auflage), Pearson 2007
- Coulouris, George et al: Verteilte Systeme (3. Auflage), Pearson 2005
- Bengel, Günther: Grundkurs Verteilte Systeme (3. Auflage), vieweg 2004

IFB1121 Wahrscheinlichkeitsrechnung und Numerik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Jochen Staudacher
Dozent(en):	Prof. Dr. Jochen Staudacher
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.), Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Erfolgreiche Teilnahme an den Übungen, schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungs- nachweis ist Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfs- mittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung können die Studierenden

- wichtige Grundbegriffe der Wahrscheinlichkeitsrechnung und der Numerik definieren
- wichtige Verteilungen auf einfache fachspezifische Fragestellungen anwenden
- die Funktionsweise zentraler numerischer Algorithmen beschreiben
- Methoden zur Beurteilung der Komplexität, Stabilität und Genauigkeit numerischer Algorithmen in einfachen Fällen einsetzen

Lehrinhalte:

- Grundbegriffe der Wahrscheinlichkeitsrechnung
- Zufallsexperiment und Wahrscheinlichkeit
- Zufallsvariable, diskrete und stetige Verteilungen; Grenzwertsätze
- Kombinatorik
- Hypothesentest
- Zahldarstellung und Fehlerrechnung
- Stabilität und Komplexität numerischer Algorithmen
- Grundbegriffe der Numerischen Linearen Algebra
- Approximation, Interpolation und numerische Integration
- Iterative Verfahren zur Bestimmung von Nullstellen und Fixpunkten

Literatur:

- W. Dürr, H. Mayer: Wahrscheinlichkeitsrechnung und Statistik, Hanser Verlag, 8. Auflage, 2018
- M. Sachs: Wahrscheinlichkeitsrechnung und Statistik, Hanser Fachbuchverlag, 5. Auflage, 2018
- G. Fischer: Stochastik einmal anders: Parallel geschrieben mit Beispielen und Fakten, vertieft durch Erläuterungen, Vieweg+Teubner, 1. Auflage, 2005
- Mor Harchol-Balter. Introduction to probability for computing. Cambridge University Press, 2023.
- M. Mitzenmacher, E. Upfal: Probability and Computing, Cambridge University Press, 2. Auflage, 2017.
- Thomas Huckle, Stefan Schneider: Numerische Methoden: Eine Einführung für Informatiker, Naturwissenschaftler, Ingenieure und Mathematiker, Springer, 2. Auflage, 2006.
- Jochen Werner: Numerische Mathematik 1, Vieweg, 1992.
- Endre Süli, David Mayers: An Introduction To Numerical Analysis, Cambridge University Press, 2003.
- Lloyd N. Trefethen, David Bau III: Numerical Linear Algebra, SIAM, 1997.
- Günther Hämmerlin, Karl-Heinz Hoffmann: Numerische Mathematik, Springer, 4. Auflage, 1994.

IFB1122 Software Praktikum

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Rieck
Dozent(en):	Prof. Dr. Stefan Rieck
Modultyp:	Pflichtmodul
Voraussetzungen:	Softwaretechnik 1 und 2
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	4 SWS Betreutes Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Benoteter Praktikumsbericht
Zur Prüfung zugelassene Hilfsmittel:	ohne / keine Einschränkung, alle Hilfsmittel zugelassen

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage,

- ein Softwaretechnik-Projekt eigenständig durchzuführen
- im Team zu arbeiten und sich selbst zu organisieren
- Entwicklungs-Ergebnisse vorzustellen und ihre Lösungsansätze zu diskutieren

Lehrinhalte:

- Praktische Umsetzung aller Schritte des Software-Entwicklungsprozesses im Rahmen eines Modell-Projektes, d.h. Analyse des Anwendungsbereiches, Erstellung von Spezifikationen, Systemdesign, Unittests, Systemtests an Hand einer konkreten Aufgabenstellung aus dem Softwarebereich.
- Präsentation von Ergebnissen

Literatur:

siehe Literatur zu den Modulen Softwaretechnik 1 und 2

IFB1123 Projektmanagement / IT-Projektmanagement

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Peter Klutke
Dozent(en):	Prof. Dr. Peter Klutke
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übungen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Eine 90 minütige schriftliche Prüfung, die am Ende des Semesters erfolgt.
Zur Prüfung zugelassene Hilfsmittel:	nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage, . . .

- die Bedeutung von Projektmanagement zu erläutern,
- Projektmanagement-Standards einzusetzen,
- Abhängigkeiten im Umfeld eines Projektes zu analysieren,
- den Ablauf eines Projektes und die zugehörigen Tätigkeiten des Projektmanagements miteinander zu kombinieren,
- wesentliche Techniken des Projektmanagements sicher situativ anzuwenden,
- vorausschauende, proaktive Tätigkeiten im Projektmanagement sicher einzusetzen und
- gesellschaftliche Auswirkungen und Sozialverträglichkeit von Lösungen und Innovationen im Projektmanagement besser zu beurteilen, etwa beim ganzheitlichen Projektmanagement, im Personalmanagement (Überstunden, Burnout) oder im gegenseitigen Umgang ("Klima") im Projekt.

Lehrinhalte:

- Motivation und Definitionen für das Projektmanagement
- Projektmanagementstandard PMBoK mit Projektphasen und Wissensgebieten
- Organisationsformen und Aufgabenbereich des Projektleiters
- Problemfeldanalyse, u.a. mit Nutzwertanalyse, Marginalrendite, Balanced Scorecard
- Projektinitiierung, IT-Projektdefinition und Projektstrukturplan
- Netzplantechnik (CPM und MPM) und Einsatzmittelplanung

- Kosten-, Kommunikations- und Personalmanagement
- Risikomanagement mit Wahrscheinlichkeitsbäumen
- Projektüberwachung, Projektsteuerung und Projektabschluss
- Tailoring und Zusammenarbeit im Projektmanagement
- Projektmanagement und Vorgehensmodelle; SCRUM-Einführung

Literatur:

- Burghardt, Manfred: "Projektmanagement: Leitfaden für die Planung, Überwachung und Steuerung von Projekten"; Verlag: Publicis; Auflage: 10. überarb. u. erw. (10. Januar 2018); ISBN-13: 978-3895784729
- Project Management Institute: "A guide to the Project Management Body of Knowledge (PMBOK Guide) (German version)"; Verlag: The Stationery Office Ltd; Auflage: 6th ed., 2017 (30. Januar 2018); ISBN-13: 978-1628251883
- Timinger, Holger : „Modernes Projektmanagement: Mit traditionellem, agilem und hybridem Vorgehen zum Erfolg", Wiley-VCH; 1. Edition (12. Juli 2017), ISBN-10: 3527530487, ISBN-13: 978-3527530489

IFB1124 IT-Sicherheit

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Elmar Böhler
Dozent(en):	Prof. Dr. Elmar Böhler
Modultyp:	Pflichtmodul
Voraussetzungen:	Grundkenntnisse der Programmierung
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übungen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Erfolgreiche Teilnahme an den Übungen sowie eine schriftliche Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden lernen die wichtigsten Bedrohungen und Schwachstellen heutiger IT-Systeme kennen und erhalten einen Überblick über die gängigen Techniken, Methoden und Konzepte zur Erhöhung der IT-Sicherheit. Sie werden in die Lage versetzt, die Ursache für Sicherheitsprobleme zu verstehen, Möglichkeiten und Grenzen von Sicherheitslösungen zu bewerten, einzuschätzen und Sicherheitslösungen zur Abwehr von Bedrohungen systematisch einzusetzen. Anhand von konkreten Fallbeispielen werden die Probleme und Lösungsansätze verdeutlicht. Verschiedene Aspekte der Bedeutung der IT-Sicherheit für die Gesellschaft werden beleuchtet.

Lehrinhalte:

- Grundlegende Begriffe und Sicherheitsprobleme,
- Sicherheitsbasismechanismen (Verschlüsselung, Signatur etc.),
- Interne (Anwendungs-)Sicherheit,
- Ziele externer Angriffe,
- Zugriffs- und Nutzungskontrolle,
- Sicherheit im Netzwerk.

Literatur:

- N. Pohlmann et.al.: Der IT-Sicherheitsleitfaden
- J. Buchmann: Einführung in die Kryptographie
- Eckert, Claudia: IT-Sicherheit

IFB2101 Architektur und Betrieb kommerzieller Anwendungssysteme

Allgemeines

Modulverantwortliche(r):	Dr. Wolfgang Rother
Dozent(en):	Dr. Wolfgang Rother
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Grundkenntnisse Betriebssysteme
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	4 SWS Blockunterricht
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit Vorlesung 15 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Erfolgreiche Teilnahme an den Übungen, Leistungsnachweise in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Die Gesamtleistung der Veranstaltung setzt sich zu 20% aus den Übungen und zu 80% aus der 90 min. Klausur zusammen.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden

- kennen Typen und Architekturen von kommerziellen Anwendungssystemen
- können typische Probleme im Zusammenhang mit kommerziellen Anwendungssystemen lösen

Lehrinhalte:

Typen, Architektur und Betrieb von Anwendungssystemen
Typische Problemstellungen und Lösungsansätze bzgl.

- Sicherheit
- Verfügbarkeit
- Anwendungs- und Serverkonsolidierung
- Integration von Anwendungen in existierende Umgebungen
- Systemmanagement

Literatur:

- Dietmar Abts, Wilhelm Müller: Grundkurs Wirtschaftsinformatik: Eine kompakte und praxisorientierte Einführung, ISBN: 3-8348-0596-3
- Andrew S. Tanenbaum: Modern Operating Systems 2th Edition, ISBN: 0-13-092641-8 (Deutsche Übersetzung ISBN: 3-8273-7019-1)

-
- Abraham Silberschatz, Peter Baer, Galvin Greg Gagne: Operating System Concepts, 7th Edition, ISBN: 0-471-69466-5
 - Frank G. Soltis, Fortress Rochester - The Inside Story of the IBM iSeries, ISBN: 1-58304-083-8

IFB2102 Mikrocomputertechnik mit Praktikum

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Daniel Güldenring
Dozent(en):	Prof. Dr. Daniel Güldenring
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Programmieren von IT-Systeme in C, Kenntnisse zu Rechnerarchitekturen
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Praktikum
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Die Endnote ergibt sich zu 100 % aus einer schriftlichen Prüfung (90 Minuten). Erfolgreiche Teilnahme am Praktikum ist Voraussetzung für die Zulassung zur schriftlichen Prüfung.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, einseitig beschrieben, keine Kopie, nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden

- kennen den prinzipiellen Aufbau und die Komponenten eines Mikrocomputers
- können hardware-nahe Software in C und Assembler entwickeln
- kennen die Stufen des Software-Tests und den Einsatz eines leistungsstarken Debugger

Lehrinhalte:

- Prozessor (Architekturen, Funktionselemente und Arbeitsweise)
- Programmierung eines Mikrocontrollers in Assembler und der Programmiersprache C
- Bussysteme
- Speicher (Technologien, Organisation)
- Peripheriekomponenten wie z.B. Parallelports, synchrone/asynchrone Schnittstellen, Timer-Bausteine, AD-/DA-Wandler, usw.
- DMA-Bausteine
- Watchdogsystem
- Interruptsystem und Interruptbehandlung

Literatur:

- U. Brinkschulte, T. Ungerer, (2010). Mikrocontroller und Mikroprozessoren. Heidelberg: Springer.
- M. Menge, (2005). Moderne Prozessorarchitekturen: Prinzipien und ihre Realisierungen. Berlin: Springer.
- K. Wüst, (2009). Mikroprozessortechnik: Grundlagen, Architekturen, Schaltungstechnik und Betrieb von Mikroprozessoren und Mikrocontrollern. Wiesbaden: Vieweg + Teubner.
- A. Böttcher, (2007). Rechneraufbau und Rechnerarchitektur. Berlin: Springer.
- C. Martin, (2003). Einführung in die Rechnerarchitektur: Prozessoren und Systeme. München: Fachbuchverlag Leipzig im Carl-Hanser-Verlag.
- H. Kopetz, (2011). Real-time systems: Design principles for distributed embedded applications. New York: Springer.
- Zusätzliches Lernmaterial ist im begleitenden Moodle-Kurs verfügbar

IFB2104 Administration von Rechnernetzen

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Arnulf Deinzer
Dozent(en):	Prof. Dr. Arnulf Deinzer / Dr. Dietmar Prestel
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	jedes Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Pflicht zur aktiven Teilnahme in Übungen/Praktikum, Schriftliche Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage, eigenverantwortlich als Administrator im RZ eines mittelständischen Unternehmens zu arbeiten und beherrschen da insbesondere die Möglichkeiten von MS AD (Active Directory).

Lehrinhalte:

Bereitstellung und Verwaltung von Windows Server 2012

- Windows Server 2012: Rollen und Features
- Managementtools
- Installationsoptionen
- Konfiguration von Windows Server 2012
- Einführung in Windows PowerShell

Einführung in Active Directory Domain Services (AD DS)

- AD DS-Infrastruktur
- Installation und Konfiguration von Domaincontrollern

Verwaltung von Active Directory Domain Services-Objekten

- Verwalten von Benutzerkonten
- Verwalten von Gruppenkonten
- Verwalten von Computerkonten
- Delegierung der Administration

Automatisierung der Active Directory Domain Services-Administration

- Verwendung von Kommandozeilentools
- Verwendung von Windows PowerShell

Implementierung von DHCP

- Installation einer DHCP-Serverrolle
- Konfiguration von DHCP-Scopes
- Verwalten einer DHCP-Datenbank
- Absichern und Überwachen von DHCP

Implementierung von DNS

- Namensauflösung für Windows-Clients und -Server
- Installation und Verwaltung eines DNS-Servers
- Konfiguration Active Directory-integrierter DNS-Zonen

Implementierung von Gruppenrichtlinien

- Verarbeitung von Gruppenrichtlinien
- Implementierung eines zentralen Speichers für administrative Templates

Absichern von Windows-Servern mit Gruppenrichtlinienobjekten

- Sicherheitseinstellungen
- Einschränken der Verwendung von Software
- Konfiguration der Windows-Firewall mit erweiterter Sicherheit

Literatur:

Zum Vergleich:

<http://it-training.netlogix.de/seminare/microsoft/windows-server>

IFB2106 Logik

Allgemeines

Die Logik ist ein wesentliches Werkzeug menschlichen Denkens. Gerade in der Informatik ist es wichtig, die wesentlichen Aspekte dieses Werkzeugs zu benennen und zu sehen wie man sie formalisieren und automatisieren kann. Die Veranstaltung geht auf diese Aspekte ein und zeigt ihre Grenzen auf, welche von ganz grundsätzlicher Natur sind und Rückschlüsse auch auf das prinzipielle Denkvermögen von Menschen erlauben.

Modulverantwortliche(r):	Prof. Dr. Elmar Böhler
Dozent(en):	Prof. Dr. Elmar Böhler
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Programmieren, Theoretische Informatik, Algorithmen und Datenstrukturen
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht an den Übungen; schriftliche Prüfung (90 Minuten) am Ende des Semesters; erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzung
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden können logische Aussagen erkennen und formalisieren. Sie erkennen inwiefern Formeln und Schaltkreise Codierungen von Wahrheitswertetabellen sind und werden sich der mitunter einhergehenden kombinatorischen Explosion bewusst. Sie verstehen den Unterschied und Zusammenhang zwischen Formeln und Booleschen Funktionen. Sie lernen wesentliche Familien von BF zu erkennen und vollständige Mengen von BF zu identifizieren. Die Begriffe des Folgerns und Schließens werden konkretisiert, wobei klar wird warum jede Aussage, auf die geschlossen werden kann, auch nach mechanischen Prinzipien automatisch abgeleitet werden kann (Vollständigkeitssatz). Die Studierenden lernen die Grenzen des logischen Folgerns kennen (Unvollständigkeitssatz). Sie werden in Mechanismen des automatischen Folgerns eingeführt, mit entsprechenden Werkzeugen bekannt gemacht und lernen ihre Grenzen kennen.

Lehrinhalte:

- Aussagenlogik
- Formulieren logischer Aussagen

- Formeln und Schaltkreise
- Wahrheitswertetabellen und Boolesche Funktionen
- Abgeschlossene Mengen von BF und Postsches Vollständigkeitskriterium
- Äquivalenz und Normalformen von Formeln
- Folgern und Schließen
- Vollständigkeitssatz/Unvollständigkeitssatz
- Hornformeln
- Resolution
- Prädikatenlogik
- Normalformen
- Unifikation
- Termersetzungssysteme/Automatische Beweissysteme
- Logikprogrammierung, Prolog
- Zusammenhang von Größe der WW-Tabelle und der Schaltkreisgröße
- Komplexitätsbetrachtungen z.B. der Probleme der Schaltkreisauswertung und der Schaltkreiserfüllbarkeit

Literatur:

- U. Schöning: Logik für Informatiker
- K. Wagner: Theoretische Informatik
- S. Russel, P. Norvig: Artificial Intelligence
- H. Vollmer: Introduction to Circuit Complexity

IFB2107 Softwareentwicklung für Smartphones

Allgemeines

Modulverantwortliche(r):	Dr. Dietmar Prestel
Dozent(en):	Dr. Dietmar Prestel
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Medizininformatik (B.Sc.)
Angebot und Dauer:	jedes Semester
Lehrformen:	4 SWS Seminaristischer Unterricht
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit-Unterricht 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden sind in der Lage eine IDE (Integrated Development Environment des Android Studio) zum Programmieren, übersetzen und zur Fehlersuche zu benutzen.
- Die Studierenden können mobile Apps konzipieren und sie umsetzen.
- Die Studierenden haben einen Überblick über User Interfaces von Smartphones und können Datenübertragungstechniken gezielt für gestellte Aufgabenstellungen anwenden.

Lehrinhalte:

- Android Studio
- Apps auf virtueller und realer Hardware testen
- Lebenszyklus einer Aktivität
- Ressourcen
- Intent und Intentfilter
- Multithreading
- Dienste
- Fragmente
- Datenbankverwaltung mit ROOM und SQLITE
- Contentprovider nutzen und anwenden

Literatur:

- Gargenta M., Einführung in die Android-Entwicklung, 2011 O'Reilly Verlag GmbH & Co. KG

Künneth T., Android 8 Das Praxisbuch für Java-Entwickler, 2018 Rheinwerk Computing, ISBN 978-3-8362-6058-9

IFB2109 Dokumentation und Informationssysteme

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Katja Bochtler
Dozent(en):	Prof. Dr. Katja Bochtler
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.), Studiengang Medizininformatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	In diesem Modul wird die Prüfungsstudienarbeit im laufenden Semester angefertigt bzw. durchgeführt. Die Prüfungsstudienarbeit kann dabei aus schriftlichen Ausarbeitungen, Präsentationen, Arbeiten am PC oder der Bearbeitung von Aufgabenstellungen im Rahmen einer Projektarbeit und einem Abschlussbericht bestehen. Der Arbeitsumfang beträgt 40 - 60 Stunden.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage ...

- das Prinzip der systematischen Planung medizinischer Dokumentationssysteme und wichtige Prinzipien für die Gestaltung medizinischer Ordnungssysteme zu verstehen.
- die Rolle der Dokumentation für die Medizinische Informatik zu verstehen.
- zu beschreiben wie ein Krankenhaus (KH) aufgebaut ist und welche Aufgaben es hat.
- die Herausforderungen dieser Aufgaben an die Informationsverarbeitung zu nennen.
- die Bedeutung der Informationsverarbeitung für ein KH zu erklären.
- ein Krankenhausinformationssystem zu definieren.
- Werkzeuge der Informationsverarbeitung im KH sowie ihre Aufgaben zu benennen.
- Probleme zu erkennen, die die heterogene IT-Infrastruktur im KH sowie im Gesundheitswesen allgemein verursachen.
- zu erklären welche Konzepte der Integration im Gesundheitswesen existieren.

Lehrinhalte:

Grundlagen der folgenden Themen werden mit einem vertieften praktischen Teil vermittelt:

- Dokumentation in der Medizin als allgegenwärtige, unterstützende Aufgabe
- Grundbegriffe zu Informationssystemen im Gesundheitswesen (ISG)
- Medizinische Ordnungssysteme und Terminologien
- Zusammenhang zwischen medizinischer Dokumentation und ISG
- Grundlagen zum Management von ISG
- Aufgaben von Krankenhausinformationssystemen
- Definition und Aufgaben von elektronischen Patientenakten
- Grundlagen der Informationsverarbeitung in Gesundheitsnetzwerken

Literatur:

- Dugas: Medizininformatik. Ein Kompendium für Studium und Praxis; Springer Vieweg; 2017
- Ammenwerth, Haux, Knaup-Gregori, Winter: IT-Projektmanagement im Gesundheitswesen; Schattauer; 2014
- Leiner, Gaus, Haux: Medizinische Dokumentation; Schattauer; 2011.
- Johnner, Haas: Praxishandbuch IT im Gesundheitswesen; Hanser; 2009.

IFB2110 Digitale Medien

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Klaus Ulhaas
Dozent(en):	Prof. Dr. Klaus Ulhaas
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Programmierkenntnisse, mathematische Grundlagen
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Anwesenheitspflicht in den Übungen, Leistungsnachweise in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfsmittel:	nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage

- die Bedeutung und Relevanz Digitaler Medien und deren Kompressionsverfahren für Computer Games und Browser-Anwendungen einzustufen.
- Abtastungsprobleme bei Digitalen Medien zu erkennen und die Ursache hierfür zu identifizieren.
- den Informationsgehalt einer Nachricht zu bestimmen.
- verlustfreie Kompressionsverfahren zu benennen und die Verfahren auf unterschiedliche Nachrichten wie z.B. Bild, Ton oder Textinformationen anzuwenden.
- zu verschiedenen Kodierungen anzugeben, wie gut die jeweilige erreichte Kompression ist.
- eine Lauflängenkodierung, Huffman-Kodierung, arithmetische Kodierung, LZW-Kodierung anzuwenden und günstige und ungünstige Anwendungsgebiete zu benennen.
- Verlustbehaftete Kompressionsverfahren zu benennen.
- die Verfahren JPEG und MPEG-Kodierung anzuwenden und teilweise zu implementieren.
- Verfahren zum Schutz digitaler Medien zu benennen und einfache Verfahren programmiertechnisch umzusetzen.

Lehrinhalte:

- Begriffsdefinitionen und wirtschaftliche Sicht zu Digitalen Medien und Digitalen Gütern
- Motivation für den Einsatz Digitaler Medien in Games und Browseranwendungen
- Abtastung und Digitalisierung
- Informationstheoretische Grundlagen zum Verständnis der Kompressionstechnik
- Verlustfreie Kompressionsverfahren
- Verlustbehaftete Kompressionsverfahren
- Urheberrecht und Mediensicherheit

Literatur:

Literatur und Empfehlungen zu Einzelthemen werden fortlaufend in der Vorlesung bekanntgegeben

IFB2115 Operations Research

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Jochen Staudacher
Dozent(en):	Prof. Dr. Jochen Staudacher
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.), Studiengang Medizininformatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung in kleinen Gruppen (14tägig 90 Minuten)
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit Vorlesung 15 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Erfolgreiche Teilnahme an den Übungen, Leistungsnachweise zu Praktikumsaufgaben in den Übungen, schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung können die Studierenden

- die grundlegenden Konzepte der Linearen Optimierung definieren
- einfache betriebswirtschaftliche Aufgaben als mathematische Modelle formulieren
- Methoden der Linearen Optimierung auf einfache Fragestellungen aus der Informatik und den Wirtschaftswissenschaften anwenden und die Ergebnisse interpretieren
- einfache Sensitivitätsanalysen durchführen
- die Grenzen der Linearen Optimierung exemplarisch aufzeigen

Lehrinhalte:

- Mathematische Grundlagen der Linearen Optimierung
- Das Simplex-Verfahren und seine Varianten
- Dualitätstheorie
- Alternativen zum Simplex-Verfahren
- Spezialfälle der Linearen Optimierung
- Ganzzahlige lineare Optimierung

- Sensitivitätsanalysen, Parametrische Lineare Optimierung
- Einfache Zweipersonen-Nullsummenspiele

Literatur:

- A. Koop, H. Moock: Lineare Optimierung: Eine anwendungsorientierte Einführung in Operations Research, Spektrum Akademischer Verlag, 2. Auflage, 2018
- P. Stingl: Operations Research: Lineare Optimierung, Hanser Fachbuchverlag, 1. Auflage, 2002
- H.-J. Zimmermann: Operations Research: Methoden und Modelle. Für Wirtschaftsingenieure, Betriebswirte, Informatiker, Vieweg+Teubner, 2. Auflage, 2007
- J. Schwarze: Mathematik für Wirtschaftswissenschaftler. Band 3: Lineare Algebra, Lineare Optimierung und Graphentheorie, Nwb Verlag, 13. Auflage, 2011
- K. Neumann, M. Morlock: Operations Research, Hanser Fachbuchverlag, 2. Auflage, 2002
- P.R. Thie, G.E. Keough: An Introduction to Linear Programming and Game Theory, 3rd Edition, 2008.
- R.J. Vanderbei: Linear Programming, Springer, 4th Edition, 2014.

IFB2125 IT-Management

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Peter Klutke
Dozent(en):	Prof. Dr. Peter Klutke
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.), Studiengang Medizininformatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Praktikum in kleinen Projektteams
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	90 Minuten schriftliche Prüfung am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage, ...

- die Grundprinzipien des IT-Managements und der IT-Governance zu erläutern,
- die Prozesse des ITIL-Frameworks zum IT Service Management zu erklären und auf Fallbeispiele anzuwenden,
- bestehende IT-Services zu analysieren oder neue IT-Services zu entwickeln und
- gesellschaftliche Auswirkungen und Sozialverträglichkeit von Lösungen und Innovationen im IT-Management besser zu beurteilen, etwa in Bezug auf die Arbeitsbelastung (Ressourcenmanagement, Überstunden, Burnout) oder die Generierung von (Mehr-)Werten durch IT-Services (Utility, Warranty).

Lehrinhalte:

- Grundprinzipien des IT-Managements und der IT-Governance
- IT-Service Management nach ITIL (IT Infrastructure Library), erläutert am durchgehenden Beispiel eines IT-Service-Providers im Gesundheitswesen
- Besprechung von Fallbeispielen zur Veranschaulichung des ITIL-Prozessmodells
- Beispielhafte Analyse und Entwicklung von IT-Services

Literatur:

- Beims, Martin und Ziegenbein, Michael: "IT-Service-Management in der Praxis mit ITIL®: Der Einsatz von ITIL® Edition 2011, ISO/IEC 20000:2011, COBIT® 5 und PRINCE2®", Verlag: Carl Hanser Verlag GmbH & Co. KG; Auflage: 4., überarbeitete und

erweiterte (4. Dezember 2014), ISBN-13: 978-3446441378

- Tiemeyer, Ernst: "Handbuch IT-Management: Konzepte, Methoden, Lösungen und Arbeitshilfen für die Praxis", Verlag: Carl Hanser Verlag GmbH & Co. KG; Auflage: 6., überarbeitete und erweiterte (13. Februar 2017), ISBN-13: 978-3446443471
- Johannsen, Wolfgang und Goeken, Matthias: "Referenzmodelle für IT-Governance: Methodische Unterstützung der Unternehmens-IT mit COBIT, ITIL & Co", Verlag: dpunkt Verlag; Auflage: 2. Aktual. (29. November 2010), ISBN-13: 978-3898646161

IFB2129 Grundlagen der Digitalen Produktion

Allgemeines

Informatik ist in der Produktion heute nicht mehr wegzudenken und in der Digitalen Produktion ein besonders spannendes Anwendungsgebiet, in dem Informatik sichtbar und erlebbar ist: z.B. durch hunderte von Industrierobotern im Rohbau einer Automobilproduktion, Digitale 3D-Modelle der Fabrik für den Auf- oder Umbau eines neuen Produkts oder 3D-Stücklisten zur Abbildung tausender Fahrzeugvarianten.

Die Digitale Produktion ist ein interdisziplinäres Fachgebiet, bei dem Informatik auf Maschinenbau, Elektrotechnik oder Wirtschaftsingenieurwesen trifft. Die Vorlesung vermittelt die Grundlagen, damit sie auf Augenhöhe mit anderen Disziplinen zusammenarbeiten und das Thema Industrie 4.0, Digitale Zwillinge oder das Industrial Metaverse in der Praxis weiterentwickeln können.

Modulverantwortliche(r):	Prof. Dr. Bernd Lüdemann-Ravit
Dozent(en):	Prof. Dr. Bernd Lüdemann-Ravit / N.N. Lehrbeauftragter
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	ca. 40 Stunden Präsenzzeit Vorlesung ca. 20 Stunden Präsenzzeit Übungen ca. 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Die Endnote ergibt sich zu 100 % aus einer schriftlichen Prüfung (90 Minuten).
Zur Prüfung zugelassene Hilfsmittel:	ohne / keine Einschränkung, alle Hilfsmittel zugelassen, Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden kennen die Grundlagen der Digitalen Produktion und verstehen, welche Methoden, Prozesse und Systeme zur Modellierung, Programmierung, Simulation und den operativen Betrieb einer Produktion zum Einsatz kommen. Im Einzelnen sind die Studierenden nach erfolgreicher Beendigung in der Lage:

- Wichtige Kenngrößen und den Aufbau einer Produktion zu verstehen.
- Vor und nachgelagerte Unternehmensprozesse einer Produktion zu kennen.
- Techniken zur Modellierung von Unternehmensprozessen zu kennen und anzuwenden.
- Möglichkeiten des praxisrelevanten Einsatzes von Digitalen Methoden zur Planung und Absicherung im Produktentstehungsprozesses zu kennen.
- Einfache 3D-Geometriemodelle für CAD kennen und selbstständig aufzubauen.

- Aufbau und Verwendung von Stücklisten und einfache Schaltpläne zu verstehen.
- Datenmodelle (Produkt, Prozess und Ressource) der Digitalen Fabrik zu konzipieren.
- IOTT-Kommunikationsprotokolle (OPC UA, MQTT) zu verstehen.
- Wichtige Mechanismen von Feldbusprotokollen (z.B. EtherCAT) zu verstehen.
- Aufgaben von Steuerungssysteme (ERP, MES; PLC) der Produktion zu erklären.
- Grundlagen von Industrierobotern (u.a. Lage im Raum, Bahnplanung) zu verstehen bzw. anzuwenden
- Einfache Programme für Industrieroboter und SPS zu schreiben
- Wesentliche Mechanismen von IIOT-Kommunikationsprotokolle (OPC UA, MQTT) zu verstehen.

Lehrinhalte:

Die Vorlesung gibt einen Überblick und erste Einblicke in die Digitale Produktion. Im Einzelnen werden folgende Themen behandelt:

- Grundlagen der Produktionswirtschaft (z.B. OEE) und wesentliche Unternehmensprozesse
- Modellierung von Unternehmensprozessen
- Modellierung von Produkten und Fabriken (Stückliste, CAD, Elektroplanung)
- Digitalen Methoden zur Planung und Absicherung im Produktentstehungsprozesses
- Grundlagen Industrierobotertechnik
- Funktionsweise von Feldbussen (z.B. EtherCAT) und speicherprogrammierbarer Steuerungen (SPSen)
- Mechanismen von IIOT-Kommunikationsprotokollen (MQTT; OPC UA)
- Aufgaben von überlagerten Steuerungen (MES, ERP)

Literatur:

Folgende Literatur wird zur Vertiefung empfohlen, wird aber nicht vorausgesetzt

- Kellner, Lienhard, Lukesch: Produktionswirtschaft – Planung, Steuerung und Industrie 4.0, Springer 2018
- Huber: Industrie 4.0 in der Automobilproduktion, Springer Vieweg 2018
- Kletti, J.: Lehrbuch für digitales Fertigungsmanagement Manufacturing Execution Systems MES, Springer, 2021
- Schleipen: Praxishandbuch OPC UA, 1. Auflage, Vogel Business, Media GmbH, 2020
- Schnell: Bussysteme in der Automatisierung und Prozesstechnik, 9. Auflage, Springer, 2019
- Weber: Industrieroboter, Hanser, 2019
- Bracht, Geckler, Wenzel: Digitale Fabrik, Methoden und Praxisbeispiele, Springer 2011

IFB2130 Grundlagen von eHealth

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Matthias Becker
Dozent(en):	Prof. Dr. Matthias Becker
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Dokumentation und Informationssysteme
Verwendbarkeit:	Studiengang Medizininformatik (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.), Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	In diesem Modul wird die Prüfungsstudienarbeit im laufenden Semester angefertigt bzw. durchgeführt. Die Prüfungsstudienarbeit kann dabei aus schriftlichen Ausarbeitungen, Präsentationen, Arbeiten am PC oder der Bearbeitung von Aufgabenstellungen im Rahmen einer Projektarbeit und einem Abschlussbericht bestehen. Der Arbeitsumfang beträgt 40 - 60 Stunden.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Nach erfolgreicher Beendigung der Veranstaltung haben die Studierenden einen Überblick über die aktuelle Situation der Gesundheitsversorgung und können die Notwendigkeit einer IT-Unterstützung der Prozesse im Gesundheitswesen erklären.
- Die Studierenden kennen verfügbare und zukünftige Anwendungen von eHealth, um optimal Geschäfts- und klinische Prozesse im Gesundheitswesen zu unterstützen.
- Die Studierenden sind in der Lage die Anforderungen, inklusive regulatorischer Art, an den IT-Systemen im Rahmen des Gesundheitswesens zu verstehen.
- Die Studierenden haben einen Überblick über aktuelle und zukünftige Entwicklungen der IT im Gesundheitswesen.
- Die Studierenden können über die möglichen Auswirkungen der Digitalisierung des Gesundheitswesens auf das Individuum und die Gesellschaft reflektieren.
- Die Studierenden sind in der Lage semantische Standards der Medizininformatik zu verstehen und anzuwenden (ICD, OPS, SNOMED CT, LOINC)
- Die Studierenden sind in der Lage syntaktische Standards der Medizininformatik zu verstehen und anzuwenden (HL7, DICOM, CDA, openEHR)

Lehrinhalte:

- Grundlagen des Gesundheitswesens und Motivation für den Einsatz von IT-Systemen
- Ziele und Einsatzbereiche von eHealth sowie Prozesse der Kommunikation (IHE)
- Beispielanwendungen von eHealth: eGK, eRezept, eMedikation, Gesundheitsportal, Telemedizin, eDokumentation
- Gesundheitstelematik: Akteure, Komponenten, Protokolle und Standards (HL7 FHIR, CDA, DICOM)
- Entwicklung von eHealth in Deutschland: eHealth-Gesetz, regulatorischer Rahmen
- Aktuelle Entwicklungen: DiGAs, DiPAs, data-driven health, precision medicine, mHealth
- Ethische und gesellschaftliche Aspekte von eHealth: Gerechtigkeit, Zugang, Digital divide, Datenschutz
- Semantische Interoperabilität (ICD, OPS, SNOMED CT, LOINC)

Literatur:

- Robin Haring (Hrsg.), Gesundheit Digital , Springer 2019
- Johannes Jörg, Digitalisierung in der Medizin, Springer 2018
- Florian Fischer, Alexander Krämer (Hrsg.), eHealth in Deutschland, Springer 2016
- Volker P. Andelfinger, Till Hänisch (Hrsg.), eHealth - Wie Smartphones, Apps und Wearables die Gesundheitsversorgung verändern werden, Springer 2016
- Jähn, K. Und Nagel, E., eHealth, Springer 2014, 978-3642639319
- Haas, P., Gesundheitstelematik, Springer 2006, 978-3540207405
- Trill, R. (Hrsg.), Praxisbuch eHealth, Kohlhammer 2009, 978-3-17-019988-0
- Aktuelle Publikationen zu den oben genannten Themen

IFB2131 Text Mining und Information Extraction

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Matthias Becker
Dozent(en):	Prof. Dr. Matthias Becker
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	
Verwendbarkeit:	Studiengang Medizininformatik (B.Sc.), Studiengang Informatik (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	In diesem Modul wird die Prüfungsstudienarbeit im laufenden Semester angefertigt bzw. durchgeführt. Die Prüfungsstudienarbeit kann dabei aus schriftlichen Ausarbeitungen, Präsentationen, Arbeiten am PC oder der Bearbeitung von Aufgabenstellungen im Rahmen einer Projektarbeit und einem Abschlussbericht bestehen. Der Arbeitsumfang beträgt 40 - 60 Stunden. Endnotenbildender Leistungsnachweis in diesem Modul: - Annotation und Analyse von deutschsprachigen Dokumenten - Implementierung einer Machine Learning Pipeline - Ausarbeitung über die Ergebnisse und Metriken der Pipeline - Kurzpräsentation der Ergebnisse
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Im Rahmen der Vorlesung Text Mining und Information Extraction, werden den Studierenden die Grundlagen im Bereich Datenanalyse und Textmining vermittelt. Dieses Wissen soll auf anonymisierte medizinische Textdokumente angewandt werden. Ziel soll es sein, die unterschiedlichen Analysemethoden zu verproben und zu bewerten. Die zur Verfügung gestellten Dokumente werden durch die Studierenden vorverarbeitet, so dass diese durch Textmining-Werkzeuge verarbeitet werden können. Anschließend werden die vorverarbeiteten Texte mit verschiedenen Standard-Werkzeugen analysiert. Dabei werden verschiedene medizinische Klassifikationen zur Abbildung der Informationen verwendet. Ziel dieser Analyse ist es zu bewerten, wie gut und genau die Textmining Werkzeuge die

Informationen aus den Dokumenten extrahieren können und inwieweit eine nicht maschinelle Nachverarbeitung der Ergebnisse notwendig ist.

Lehrinhalte:

- Einführung in Data Mining, Textmining, Machine Learning, Natural Language Processing und Information Extraction
- Einführung in die Datenanalyse von deutschsprachigen Dokumenten
- Übersicht über Tools aus dem Bereich der Textanalyse
- Übersicht von NLP Modellen und NLP Korpusse (national und international)
- Probleme und Herausforderungen von Textmining
- Kennzahlen für die Qualität von NLP Pipelines
- Stand der Wissenschaft im Bereich NLP
- Anwendung von klinischen Terminologien, Ontologien und Terminologiesammlungen (UMLS)
- Machine Learning mit Python (spacy.io) und LLM sowie GPT

Literatur:

- Bastian Buch, "Text Mining: Zur automatischen Wissensextraktion aus unstrukturierten Textdokumenten", Taschenbuch, 8. April 2008
- Aman Kedia, Mayank Rasu, "Hands-On Python Natural Language Processing: Explore tools and techniques to analyze and process text with a view to building real-world NLP applications", 2020
- Ashok N. Srivastava, Mehran Sahami, "Text Mining: Classification, Clustering, and Applications", Chapman & Hall/CRC Data Mining and Knowledge Discovery Series Book 10, 2009
- Chris Biemann, Gerhard Heyer, Uwe Quasthoff, "Wissensrohstoff Text: Eine Einführung in das Text Mining", Taschenbuch, 2022
- spacy.io

IFB2132 Advanced Embedded Systems

Allgemeines

Die Lehrveranstaltung vermittelt vertiefte Kenntnisse und erweiterte Fähigkeiten auf dem Gebiet der eingebetteten Systeme. Der Fokus der Lehrveranstaltung liegt dabei sowohl auf fortgeschrittenen Debugging-Methoden als auch auf der Funktionsweise und Anwendung von Echtzeit-Betriebssystemen.

Modulverantwortliche(r):	Prof. Dr. Daniel Güldenring
Dozent(en):	Prof. Dr. Daniel Güldenring
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	IFB1109 Rechnerarchitektur, IFB1108 IT-Systeme, IFB1103 Programmieren 1
Verwendbarkeit:	
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	Vorlesung: 30h Praktikum/Übung: 30h Selbststudium: 60h
Leistungsnachweis und Prüfung:	Die Endnote ergibt sich zu 100 % aus einer schriftlichen Prüfung (90 Minuten).
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, einseitig beschrieben, keine Kopie, nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden kennen fortgeschrittene Debugging-Methoden und können diese effektiv zum Debuggen eingebetteter Systeme einsetzen.
- Sie verstehen wie die JTAG-Schnittstelle, openOCD und GDB für das Debugging eingebetteter Systeme eingesetzt werden.
- Sie verfügen über vertiefte theoretische Kenntnisse zu den Grundlagen von Echtzeitsystemen.
- Sie können das Echtzeitbetriebssystem FreeRTOS für die Entwicklung von Echtzeitsystemen einsetzen.

Lehrinhalte:

- Aufbau und Funktion der JTAG-Schnittstelle
- Fortgeschrittene Debugging-Methoden wie z. B. (bedingte) Break- und Watchpoints
- Debuggen eingebetteter Systeme mit GDB über OpenOCD und JTAG
- Echtzeit-Betriebssysteme (Definition, Hauptkomponenten, Funktionsweise)
- Taskmodell und Nebenläufigkeit

- Scheduling-Verfahren und Feasibility Tests
- Ressourcen (Zugriff auf geteilte / begrenzte Ressourcen, kritische Abschnitte)
- Ressourcenzuordnungsprotokolle (non-preemptive critical sections, priority inheritance protocol)
- Entwickeln von Echtzeitanwendungen mit dem Echtzeitbetriebssystem FreeRTOS

Literatur:

- R. Stallman, et al., (2018). Debugging with GDB; Boston: Free Software Foundation.
- E. Kienzle, J. Friedrich, (2008). Programmierung von Echtzeitsystemen; München: Hanser Verlag
- G. C. Buttazzo, (2011). Hard Real-Time Computing Systems; Boston: Springer.
- H. Kopetz, (2011). Real-Time Systems; Boston: Springer.
- D. Zöbel, (2020). Echtzeitsysteme; Berlin: Springer.
- Jane W. S. Lui, (2000). Real-Time Systems, New Jersey: Prentice Hall
- R. Barry (2016): Mastering the FreeRTOS™ Real Time Kernel; Real Time Engineers Ltd.
- A. Robbins, (2005). GDB Pocket Reference; Sebastopol: O'Reilly Media.
- Zusätzliches Lernmaterial ist im begleitenden Moodle-Kurs verfügbar.

IFB2133 Telemedizin

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Petra Friedrich
Dozent(en):	Prof. Dr. Petra Friedrich
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Medizininformatik (B.Sc.), Studiengang Informatik (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Praktikum, Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	SPA = Studienprüfungsarbeit Die Endnote ergibt sich zu 100 % aus einer Portfolioprüfung mit Präsentation. Erfolgreiche Teilnahme an Praktika und termingerechte Abgabe einer schriftlichen Ausarbeitung.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Nach der Teilnahme an der Modulveranstaltung sind die Studierenden in der Lage, den Lerninhalt auf neue Probleme der Telemedizin anzuwenden. Sie können die Möglichkeiten und Grenzen, die sich aus den Produkten dieser Disziplin ergeben, kritisch bewerten.
- Nach der Teilnahme an der Modulveranstaltung sind die Studierenden in der Lage, moderne Methoden der Telemedizin sowie sensorgestützter telematischer Diagnose- und Therapieverfahren zu verstehen sowie Möglichkeiten und Grenzen der behandelten Methoden kritisch zu bewerten. Es wird insbesondere ein vertieftes Verständnis für die Besonderheiten von klinischen Studiendesigns und -Settings erlangt.

Lehrinhalte:

Die Verknüpfung elektronischer Medien und Systeme mit medizinischen Sensoren öffnet den Weg zu einer personalisierten telematischen Medizin. Vor diesem Hintergrund werden anhand von Beispielen aktuelle Strategien präsentiert, mit denen sich personalisierte Diagnose- und Therapiekonzepte realisieren lassen. Ein Schwerpunkt liegt auf der Diskussion geeigneter telemedizinischer Sensoren für diverse Krankheitsbilder.

- Einführung in die Thematik und das Arbeitsgebiet Telemedizin mit den dazugehörigen physiologischen, medizinischen und technischen Grundlagen
- Praktische Einführung in grundlegende Arbeitstechniken der Physiologie und Tele-

medizin sowie sensorgestützter Messverfahren und bioelektronischer Feedbacksysteme

- Einführung in das zu verwendende Messequipment
- Anwendungen für eine personalisierte und individualisierte telematische Medizin
- Wissenschaftlichen Planung, Durchführung und Auswertung von Experimenten und Testreihen
- Dokumentation, Protokollführung, statistische Datenauswertung
- Recherche und Machbarkeitsstudien für konkrete projektbezogene Aufgaben

Literatur:

Lehrmaterial ist im Hochschulnetz verfügbar, Gastdozenten (Medizintechnikhersteller), Exkursionen (Medizintechnikhersteller/-anwender, Telemedizinanbieter, Messen, usw.)

- B. Wolf, J. Gausemeier et. al., TELEMEDIZINISCHE ASSISTENZSYSTEME, Technik, Markt, Geschäftsmodelle
- Spektrum Telemedizin Bayern 2014 Bayerische TelemedAllianz Dr. Siegfried Jedamzik, ISBN 978-3-00-045501-8

IFB2135 Einführung in die Bioinformatik

Allgemeines

Dieser Kurs bietet einen Überblick über die Bioinformatik mit dem Schwerpunkt auf der Anwendung computergestützter Techniken zur Analyse und Interpretation biologischer Daten. Die Studierenden lernen die wichtigsten bioinformatischen Werkzeuge, Datenbanken und Algorithmen kennen, die in der Genomik, Proteomik und anderen Bereichen der biologischen Forschung eingesetzt werden.

Modulverantwortliche(r):	Prof. Dr. Rafael Mayoral Malmström
Dozent(en):	Prof. Dr. Rafael Mayoral Malmström
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.) und Studiengang Medizininformatik (B.Sc.) - Für Studierende mit Studienbeginn vor WS 2025/26
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	In diesem Modul wird die Prüfungsstudienarbeit im laufenden Semester angefertigt bzw. durchgeführt. Die Prüfungsstudienarbeit kann dabei aus schriftlichen Ausarbeitungen, Präsentationen, Arbeiten am PC oder der Bearbeitung von Aufgabenstellungen im Rahmen einer Projektarbeit und einem Abschlussbericht bestehen. Der Arbeitsumfang beträgt 40 - 60 Stunden.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage ...

- Die grundlegenden Konzepte und Anwendungen der Bioinformatik zu verstehen und die historische Entwicklung sowie die zentrale Rolle der Bioinformatik in der modernen biologischen Forschung zu erläutern.
- Biologische Datenbanken (z.B. GenBank, UniProt, Ensembl) gezielt zu durchsuchen und relevante Informationen strukturiert zu extrahieren.
- Verschiedene Algorithmen zur Sequenzanalyse (z.B. Needleman-Wunsch, Smith-Waterman) anzuwenden und die Ergebnisse von Paarweiser- und Multipler-Sequenzalignement-Methoden zu interpretieren.
- Genomdatensätze aus modernen Hochdurchsatzsequenzierungen (NGS) zu analysie-

ren, unter Verwendung geeigneter Softwarewerkzeuge wie BLAST, Bowtie und BWA.

- RNA-Seq-Daten für transkriptomische Analysen zu verarbeiten und verschiedene Ansätze zur quantitativen Genexpressionsanalyse zu nutzen, um differenziell exprimierte Gene zu identifizieren.
- Proteinstrukturen vorherzusagen, zu visualisieren und die bioinformatischen Methoden in der Proteomanalyse anzuwenden.
- Phylogenetische Analysen durchzuführen und phylogenetische Bäume zu erstellen, um evolutionäre Beziehungen abzuleiten.
- Strukturierte und funktionelle Bioinformatik-Werkzeuge (z.B. Jmol, Chimera) zur Analyse von Proteindaten und molekularen Strukturen anzuwenden.
- Komplexe biologische Netzwerke und Signalwege zu modellieren und dabei Systeme wie KEGG und Cytoscape zu verwenden.

Lehrinhalte:

Der Kurs deckt ein breites Spektrum an bioinformatischen Themen ab und bietet einen praxisnahen Einblick in die relevanten Werkzeuge und Methoden der modernen Bioinformatik. Die Studierenden lernen, Daten aus unterschiedlichen Quellen zu integrieren und mit bioinformatischen Algorithmen und Softwarewerkzeugen zu analysieren.

- Biologische Datenbanken: Datenbanktypen (Sequenz-, Struktur- und Funktionsdatenbanken), relevante biologische Datenbanken (GenBank, UniProt, Ensembl, PDB), Datenbankrecherche und -retrieval.
- Sequenzalignment und Analyse: DNA-, RNA- und Proteinsequenzen, Paarweise Sequenzalignment (Needleman-Wunsch, Smith-Waterman), Multiple Sequenzalignments (ClustalW, MUSCLE).
- Genomik und Genomanalyse: Grundlagen der Genomik, Hochdurchsatzsequenzierung, Genom Assemblierung und -annotation, Werkzeuge (BLAST, Bowtie, BWA).
- Transkriptomik: RNA-Sequenzierung, Genexpressionsanalyse, Identifikation differenziell exprimierter Gene (HISAT, StringTie, DESeq2).
- Proteomik und Proteinanalyse: Proteomik-Methoden, Proteinstrukturvorhersage und -modellierung, Visualisierung von Proteinstrukturen (SWISS-MODEL, PyMOL).
- Phylogenetische Analyse: Grundlagen der Evolutionsbiologie, Methoden zur Erstellung phylogenetischer Bäume (UPGMA, Neighbor-Joining, Maximum-Likelihood), Software (MEGA, PhyML).
- Strukturelle Bioinformatik: Proteinstruktur-Datenbanken (PDB, SCOP, CATH), Molekulare Visualisierungstools (Jmol, Chimera), Strukturelle Ausrichtung und Vergleich.
- Bioinformatische Algorithmen und Werkzeuge: Überblick über gängige Algorithmen, Anwendungen von Hidden Markov Models (HMMs), maschinelles Lernen in der Bioinformatik.
- Systembiologie und Netzwerkanalyse: Modellierung biologischer Netzwerke, Analyse von Signalwegen, Software (KEGG, Cytoscape).

Literatur:

- Arthur M. Lesk, Introduction to Bioinformatics, Oxford University Press, 5. Auflage (2019)
- Ranier Merkl, Bioinformatik : Grundlagen, Algorithmen, Anwendungen, Wiley, 4. Auflage (2022)
- Thomas Dandekar , Meik Kunz, Bioinformatik: Ein einführendes Lehrbuch, Springer

Spektrum, 2. Auflage (2021)

- Online Ressourcen: NCBI Bookshelf (<https://www.ncbi.nlm.nih.gov/books/>), NCBI Tutorials and Resources (<https://www.ncbi.nlm.nih.gov/home/tutorials/>), Ensembl Documentation (<https://www.ensembl.org/info/docs/index.html>), sandbox.bio

IFB2349 Maschinelles Lernen

Allgemeines

Einführung in die grundlegenden Technologien und Methoden des Maschinellen Lernens.

Modulverantwortliche(r):	Prof. Dr. Patrick Scharpfenecker
Dozent(en):	Prof. Dr. Patrick Scharpfenecker
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Programmieren, Wahrscheinlichkeitsrechnung & Numerik
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht an den Übungen; schriftliche Prüfung (90 Minuten) am Ende des Semesters; erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzung
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden ordnen das Maschinelle Lernen als Disziplin im Themengebiet der Künstlichen Intelligenz ein. Sie benennen die Unterschiede zwischen überwachtem, unbeaufsichtigtem und bestärkendem Lernen und beschreiben damit verbundenen Ansätze und Methoden.

Die Studierenden wählen geeignete Algorithmen des Maschinellen Lernens anhand der Lernaufgabe aus. Sie können die Aufgabe mit Tools lösen, die im Kontext des Maschinellen

Lernens zum Einsatz kommen. Darüber hinaus können sie Anforderungen an die benötigte Hardware formulieren.

Lehrinhalte:

- Begriffliche Einordnung des „Maschinellen Lernens“ und Abgrenzung gegenüber anderen Teilgebieten der Künstlichen Intelligenz.
- Methoden des überwachten, unbeaufsichtigten und bestärkenden Lernens.
- Grundlegende Algorithmen des maschinellen Lernens und deren Anwendung, z.B.:
 - Lineare Regression
 - Entscheidungsbäume
 - Clustering
 - Support Vector Machines (SVM)
 - Neuronale Netze
- Anwendungsbeispiele

- Tools und Bibliotheken
- Hardware für ML Anwendungen

Literatur:

- Russell, Stuart J., and Peter Norvig. Artificial intelligence: a modern approach. Pearson, 2016.
- Hastie, Trevor, et al. The elements of statistical learning: data mining, inference, and prediction. Vol. 2. New York: springer, 2009.
- Deisenroth, Marc Peter, A. Aldo Faisal, and Cheng Soon Ong. Mathematics for machine learning. Cambridge University Press, 2020.
- Alpaydin, Ethem. Maschinelles Lernen. Walter de Gruyter GmbH & Co KG, 2022.
- Frochte, Jörg. Maschinelles Lernen: Grundlagen und Algorithmen in Python. Carl Hanser Verlag GmbH Co KG, 2019.

IFB2766 Funktionale Sicherheit

Allgemeines

Einführung in Funktionale Sicherheit mit besonderer Beachtung der Anforderungen an Informatik und zur Robotik.

Modulverantwortliche(r):	Prof. Dr. Rolf Jung
Dozent(en):	Prof. Dr. Rolf Jung
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (Bachelor)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	Vorlesung: 45h Übung: 15h Selbststudium: 90h
Leistungsnachweis und Prüfung:	Die Endnote ergibt sich zu 100 % aus einer schriftlichen Prüfung (90 Minuten).
Zur Prüfung zugelassene Hilfsmittel:	Teile der Norm ISO 13849 nach Vorgabe, nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

- Studenten kennen den Begriff 'Funktionale Sicherheit' und damit verbundene Definitionen aus den Normen
- Sie kennen Managementmethoden und können diese dem Sicherheitslebenszyklus zuordnen
- Studenten können die Methode FMEDA im Sicherheitsprozess einordnen sowie Sicherheitskennzahlen bestimmen und anwenden
- Teilnehmer kennen Testmethoden bezüglich System, Hardware, Software und können aus System und Produktanforderungen Testanforderungen erstellen
- Studenten können Softwarewerkzeuge klassifizieren
- Die Studenten kennen einen Zulassungsprozess und können Dokumente für eine Sicherheitszulassung zusammenstellen
- Teilnehmer kennen die Elemente des Sicherheitsprozesses und können sie dem Entwicklungsprozess in Robotik zuordnen

Lehrinhalte:

- Einführung in die Gesetzeslage und Normenüberblick mit Begriffserklärungen
- Erläuterung der Elemente eines Sicherheitsprozesses und Erstellung eines Sicherheitsplans und Sicherheitsnachweises
- Durchführung einer Risiko- und Gefährdungsanalyse

- Ermittlung von Sicherheitskennzahlen an einem Fallbeispiel
- Durchführung einer Sicherheitsanalyse für Fehlerkombinationen
- Erarbeitung einer FMEDA mit Fallbeispiel
- Kennenlernen von Testmethoden und Erstellen eines V&V-Plans
- Klassifizierung von Softwarewerkzeugen entsprechend Sicherheitsnormen
- Analyse der Elemente und Schritte in einem Sicherheitslebenszyklus

Literatur:

- Löw, Pabst, Petry: Funktionale Sicherheit in der Praxis, dpunkt.Verlag
- Josef Börcsök, Funktionale Sicherheit: Grundzüge sicherheitstechnischer Systeme, VDE Verlag 2014

IFB2767 Einführung in Data Science

Allgemeines

Data Science ist ein interdisziplinäres Wissenschaftsfeld mit Schnittmengen zur Informatik und zur Statistik und beschäftigt sich mit der Extraktion von Wissen aus Daten. Das Ziel ist es, Erkenntnisse und Handlungsempfehlungen in konkreten Problemstellungen zu generieren und zugleich, wo möglich, das Maß der Unsicherheit in den Erkenntnissen zu quantifizieren. Die Vorlesung gibt einen breiten Überblick über die Bearbeitung und Visualisierung von Daten mit Python, und diskutiert statistische Anwendungen und Machine Learning Modelle. Grundlegende Mathematikkenntnisse werden vorausgesetzt.

Modulverantwortliche(r):	Prof. Dr. Frank Schirmeier
Dozent(en):	Prof. Dr. Frank Schirmeier
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Statistik und Wahrscheinlichkeitslehre
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweise in den Übungen, 90 minütige schriftliche Prüfung am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage, in einem Unternehmen im Gespräch mit einem Mitarbeiter der Führungs- oder Fachebene eine datenbezogene Problemstellung zu analysieren, die relevanten Daten zu identifizieren, diese in einer geeigneten Infrastruktur zu erfassen, weiterzuverarbeiten und in nachfolgenden Analyseschritten Handlungsempfehlungen zu erarbeiten.

Lehrinhalte:

Literatur:

- Jake VanderPlas: Data Science Handbook (online verfügbar)
- J. Grus: Einführung in Data Science
- Reddy, Aggarwal: Healthcare Data Analytics, 1 st ed, Chapman and Hall, 2015
- Dey, Das, Naik, Behera (Eds.): Big Data Analytics for Intelligent Healthcare Management, 1st ed, Elsevier, 2019.

IFB3100.1 Praktisches Studiensemester

Allgemeines

Modulverantwortliche(r):	Praxisbeauftragter Prof. Dr. Bernd Dreier
Dozent(en):	Praxisbeauftragter Prof. Dr. Bernd Dreier
Modultyp:	Pflichtmodul
Voraussetzungen:	Zulassung zum Vertiefungsstudium und in den Fächern des Vertiefungsstudiums mind. 20 ECTS erreicht
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	jedes Semester
Lehrformen:	Praktische Tätigkeit
Leistungspunkte:	25
Arbeitsaufwand:	Zusammenhängender Zeitraum von mind. 20 Wochen (einschließlich der praxisbegleitenden Lehrveranstaltungen), wobei die tägliche Arbeitszeit der üblichen Arbeitszeit der Ausbildungsstelle entspricht.
Leistungsnachweis und Prüfung:	Praxisbericht
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Im praktischen Studiensemester widmen sich Studierende deutlich berufsbezogenen Tätigkeiten. Die praktische Ausbildung stellt die Verbindung zwischen Studium und Berufspraxis her und ist auf die Erfordernisse in Industrie, Wirtschaft, Verwaltung etc. ausgerichtet. Die praktische Ausbildung wird durch praxisbegleitende Lehrveranstaltungen der Fakultät ergänzt. Sie dienen der Integration von Praxis und Theorie sowie der Auswertung und Vertiefung der praktischen Tätigkeiten innerhalb des praktischen Studiensemesters.

Lehrinhalte:

Literatur:

IFB3100.2A Praxisbegleitende Lehrveranstaltung Kommunizieren und Präsentieren

Allgemeines

Modulverantwortliche(r):	Hr. Grassmann
Dozent(en):	Hr. Grassmann
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Praktikum in kleinen Gruppen Blocklehrveranstaltung
Leistungspunkte:	2.5
Arbeitsaufwand:	30 Stunden Präsenzzeit 45 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Aktive Teilnahme und praktische Reflexion
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme können die Studierenden

- Sachverhalte sach- und situationsangemessen darstellen,
- verschiedene Präsentationstechniken zielgruppenorientiert anwenden,
- Argumentationsmuster und Argumentationsstrategien analysieren und beurteilen,
- sprachliche, sprecherische und rhetorische Gestaltungsmittel von Botschaften gezielt einsetzen,
- durch Stimme und Sprechweise Aufmerksamkeit/ Wirksamkeit bei Zuhörern steigern,
- aktiv mehrere Variationen körpersprachlicher Verstärkung analysieren und beherrschen
- Feedback-Prozesse in einer Gruppe organisieren und auswerten

Lehrinhalte:

- Präsentationstechniken
- Innovative Strategien und Techniken der Objektpräsentation und Selbstdarstellung

Literatur:

IFB3100.2B Praxisbegleitende Lehrveranstaltung Soziale Kompetenz

Allgemeines

Modulverantwortliche(r):	Timo Schneider
Dozent(en):	Timo Schneider
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht mit viel Übungen (Workshopcharakter)
Leistungspunkte:	2.5
Arbeitsaufwand:	30 Stunden Präsenzzeit 45 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Aktive Teilnahme und praktische Reflexion
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme können die Studierenden

- die Soziale Kompetenz als Handlungsprämisse erkennen
- ihre soziale und fachliche Kompetenz verknüpfen
- effektiv zusammenarbeiten und kooperieren

Lehrinhalte:

- 4 Kompetenzfelder eines Mitarbeiters und (zukünftigen) Führungskraft
- Kompetenz und Identität
- Selbst- und Fremdwahrnehmung
- Grundlagen der Kommunikation
- Einführung in die Projektdynamik
- Teamrollen
- Konfliktmanagement
- 1 Tag Outdoortraining

Literatur:

IFB4100 Seminar

Allgemeines

Modulverantwortliche(r):	Professoren der Fakultät
Dozent(en):	Professoren der Fakultät
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	jedes Semester
Lehrformen:	Seminar
Leistungspunkte:	5
Arbeitsaufwand:	15 Stunden Präsenzzeit Vorträge und anschließender Diskussion 135 Stunden selbständiges Arbeiten, Vorbereitung der Präsentation, Ausarbeitung der Studienarbeit
Leistungsnachweis und Prüfung:	Studienarbeit und Kolloquium
Zur Prüfung zugelassene Hilfsmittel:	ohne / keine Einschränkung, alle Hilfsmittel zugelassen

Lernergebnisse und Inhalte

Lernergebnisse:

Nach dem erfolgreichen Besuch der Lehrveranstaltung sind die Studierenden in der Lage, ...

- sich in ein vorgegebenes Thema einzuarbeiten, entsprechende Literatur zu recherchieren und aufzuarbeiten
- einen Vortrag zu einem vorgegebenen Thema zu erarbeiten, zu präsentieren und bei der nachfolgenden Diskussion zu vertreten
- eine schriftliche Ausarbeitung zu einem vorgegebenen Thema zu erstellen

Lehrinhalte:

Methoden und Vorgehensweisen zum wissenschaftlichen Arbeiten

- Verwendung der Online-Angebote der Bibliothek (Datenbanken, Kataloge, Fernleihe, Online-Zeitschriften, etc.)
- Regeln zum Zitieren (inkl. Quellenverzeichnis)
- Gestaltung von Aufbau und Gliederung (inkl. Inhaltsverzeichnis)

Literatur:

- H. Balzert, M. Schröder, C. Schäfer: Wissenschaftliches Arbeiten, W3L Verlag, 2. Auflage (2012)
- weitere Literatur in Absprache mit dem jeweiligen Dozenten

IFB5100 Projektarbeit

Allgemeines

Modulverantwortliche(r):	Professoren der Fakultät
Dozent(en):	Professoren der Fakultät
Modultyp:	Pflichtmodul
Voraussetzungen:	Software Praktikum
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	jedes Semester
Lehrformen:	Projekt
Leistungspunkte:	15
Arbeitsaufwand:	15 Stunden Präsenzzeit Unterricht 435 Stunden selbständiges Arbeiten
Leistungsnachweis und Prüfung:	Projektbericht Präsentation
Zur Prüfung zugelassene Hilfsmittel:	ohne / keine Einschränkung, alle Hilfsmittel zugelassen

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage, . . .

- ein Projekt zu planen, durchzuführen und sorgfältig zu dokumentieren
- ihr erworbenes Wissen in Standard-Anwendungsszenarien einzusetzen und erfolgreich anzuwenden
- die Entwicklungsergebnisse in Form von Postersessions und Kurzpräsentationen vorzustellen

Lehrinhalte:

projektabhängig

Literatur:

projektabhängig

IFB6100.1 Bachelorarbeit

Allgemeines

Modulverantwortliche(r):	Betreuender Professor
Dozent(en):	Betreuender Professor
Modultyp:	Pflichtmodul
Voraussetzungen:	Mind. 170 ECTS aus Basis- und Vertiefungsstudium
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	jedes Semester
Lehrformen:	-
Leistungspunkte:	12
Arbeitsaufwand:	Das Thema muss so beschaffen sein, dass die Bachelorarbeit bei zusammenhängender ausschließlicher Bearbeitung in der Regel in zehn Wochen fertiggestellt werden kann. Eine Höchstfrist von fünf Monaten darf nicht überschritten werden
Leistungsnachweis und Prüfung:	Abschlussarbeit
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Mit der Bachelorarbeit soll der Studierende beweisen, dass er in der Lage ist, eine Problemstellung - praktischer oder theoretischer Natur - innerhalb eines begrenzten und definierten Zeitraums nach wissenschaftlichen Methoden zu bearbeiten. Die Abschlussarbeit darf mit Zustimmung der Prüfungskommission in einer Einrichtung außerhalb der Hochschule ausgeführt werden.

Lehrinhalte:

Entsprechend dem gewählten Thema

Literatur:

In Absprache mit dem betreuenden Professor

IFB6100.2 Bachelorarbeit - Seminar

Allgemeines

Modulverantwortliche(r):	Betreuender Professor
Dozent(en):	Betreuender Professor
Modultyp:	Pflichtmodul
Voraussetzungen:	Mind. 170 ECTS aus Basis- und Vertiefungsstudium
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	jedes Semester
Lehrformen:	Seminar
Leistungspunkte:	3
Arbeitsaufwand:	90 Stunden selbständiges Arbeiten (incl. Präsentation)
Leistungsnachweis und Prüfung:	Vortrag über das Thema der Abschlussarbeit im Rahmen eines Seminars

Zur Prüfung zugelassene Hilfsmittel:

Lernergebnisse und Inhalte

Lernergebnisse:

Durch das Bachelorseminar sind die Studierenden in der Lage,

- sich in das Thema der Bachelorarbeit einzuarbeiten, entsprechende Literatur zu recherchieren und aufzuarbeiten
- einen Vortrag zum Thema der Bachelorarbeit zu erarbeiten, zu präsentieren und bei der nachfolgenden Diskussion zu vertreten

Lehrinhalte:

- formale Anforderungen an die Bachelorarbeit
- inhaltliche Anforderungen an Bachelorarbeit
(insbesondere auch Abstimmung mit betreuendem Professor)

Literatur:

Abhängig vom Thema der Bachelorarbeit