

Modulhandbuch
Bachelor-Studiengang
Technische Informatik

Prüfungsordnung 20.0

Wedel, den 16. Dezember 2021

Teil I

Modulhandbuch

Kapitel 1.1

Modulhandbuch

Modulverzeichnis nach Modulkürzel

B001	Analysis	28
B002	Mathematische Konzepte und Diskrete Mathematik	33
B003	Programmstrukturen 1	14
B004	Informationstechnik	19
B006	Einführung in die Digitaltechnik	23
B019	Deskriptive Statistik und Grundlagen der Linearen Algebra	63
B020	Programmstrukturen 2	42
B023	Rechnerstrukturen und Digitaltechnik	49
B032	Übertragungstechnik	60
B034	Einführung in die Betriebswirtschaft	153
B036	Programmierpraktikum	139
B037	Rechnernetze	85
B040	Algorithmen und Datenstrukturen	74
B043	Systemnahe Programmierung	80
B044	UNIX und Shell-Programmierung	55
B045	Lineare Algebra	103
B046	Ingenieurmathematik	98
B048	Elektronik	94
B052	Einführung in Datenbanken	158
B057	Fortgeschrittene Objektorientierte Programmierung	108
B058	Software-Design	135
B068	Halbleiterschaltungstechnik	123
B073	Systemtheorie	128
B085	Grundlagen der Computergrafik	131
B095	Anwendungen der Künstlichen Intelligenz	168
B097	Bildbearbeitung und -analyse	120
B099	Auslandssemester	203
B101	Echtzeitsysteme	113
B107	Einführung in die Robotik	163
B108	Großintegrierte Systeme	143
B109	Regelungstechnik	146
B111	Netzwerk- und Messtechnik	171
B118	Soft Skills	198
B133	Laborprojekt	193
B134	Diskrete Systeme	177
B135	Projekt Eingebettete Systeme	183
B143	Systementwurf mit VHDL	187
B148	Seminar Technische Informatik	150

B150 Bachelor-Thesis	214
B159 Betriebspraktikum	212
B160 Bachelor-Kolloquium.....	217
B176 Praxissemester (dual)	205
B179 Wissenschaftliche Ausarbeitung (dual).....	209
B252 Mechanik und Elektrotechnik.....	37
B254 Halbleiterphysik und Optik	69

Modulverzeichnis nach Modulbezeichnung

Algorithmen und Datenstrukturen	74
Analysis	28
Anwendungen der Künstlichen Intelligenz	168
Auslandssemester	203
Bachelor-Kolloquium	217
Bachelor-Thesis	214
Betriebspraktikum	212
Bildbearbeitung und -analyse	120
Deskriptive Statistik und Grundlagen der Linearen Algebra	63
Diskrete Systeme	177
Echtzeitsysteme	113
Einführung in Datenbanken	158
Einführung in die Betriebswirtschaft	153
Einführung in die Digitaltechnik	23
Einführung in die Robotik	163
Elektronik	94
Fortgeschrittene Objektorientierte Programmierung	108
Großintegrierte Systeme	143
Grundlagen der Computergrafik	131
Halbleiterphysik und Optik	69
Halbleiterschaltungstechnik	123
Informationstechnik	19
Ingenieurmathematik	98
Laborprojekt	193
Lineare Algebra	103
Mathematische Konzepte und Diskrete Mathematik	33
Mechanik und Elektrotechnik	37
Netzwerk- und Messtechnik	171
Praxissemester (dual)	205

Programmierpraktikum	139
Programmstrukturen 1	14
Programmstrukturen 2	42
Projekt Eingebettete Systeme	183
Rechnernetze	85
Rechnerstrukturen und Digitaltechnik	49
Regelungstechnik	146
Seminar Technische Informatik	150
Soft Skills	198
Software-Design	135
Systementwurf mit VHDL	187
Systemnahe Programmierung	80
Systemtheorie	128
UNIX und Shell-Programmierung	55
Wissenschaftliche Ausarbeitung (dual)	209
Übertragungstechnik	60

I.1.1 Erläuterungen zu den Modulbeschreibungen

Im Folgenden wird jedes Modul in tabellarischer Form beschrieben. Die Reihenfolge der Beschreibungen richtet sich nach der Abfolge im Curriculum.

Vor den Modulbeschreibungen sind zwei Verzeichnisse aufgeführt, die den direkten Zugriff auf einzelne Modulbeschreibungen unterstützen sollen. Ein Verzeichnis listet die Modulbeschreibungen nach Kürzel sortiert auf, das zweite Verzeichnis ist nach Modulbezeichnung alphabetisch sortiert.

Die folgenden Erläuterungen sollen die Interpretation der Angaben in einzelnen Tabellenfeldern erleichtern, indem sie die Annahmen darstellen, die beim Ausfüllen der Felder zugrunde gelegt wurden.

Angaben zum Modul

Modulkürzel:	FH-internes, bezogen auf den Studiengang eindeutiges Kürzel des Moduls
Modulbezeichnung:	Textuelle Kennzeichnung des Moduls
Lehrveranstaltungen:	Lehrveranstaltungen, die im Modul zusammen gefasst sind, mit dem FH-internen Kürzel der jeweiligen Leistung und ihrer Bezeichnung
Prüfung im Semester:	Auflistung der Semester, in denen nach Studienordnung erstmals Modulleistungen erbracht werden können
Modulverantwortliche(r):	Die strategischen Aufgaben des Modulverantwortlichen umfassen insbesondere: <ul style="list-style-type: none">▪ Synergetische Verwendung des Moduls auch in weiteren Studiengängen▪ Entwicklung von Anstößen zur Weiterentwicklung der Moduls und seiner Bestandteile▪ Qualitätsmanagement im Rahmen des Moduls (z. B. Relevanz, ECTS-Angemessenheit)▪ Inhaltsübergreifende Prüfungstechnik. Die operativen Aufgaben des Modulverantwortlichen umfassen insbesondere: <ul style="list-style-type: none">▪ Koordination von Terminen in Vorlesungs- und Klausurplan▪ Aufbau und Aktualisierung der Modul- und Vorlesungsbeschreibungen▪ Zusammenführung der Klausurbestandteile, die Abwicklung der Klausur (inkl. Korrekturüberwachung bis hin zum Noteneintrag) in enger Zusammenarbeit mit den Lehrenden der Modulbestandteile

- Funktion als Ansprechpartner für Studierende des Moduls bei sämtlichen modulbezogenen Fragestellungen.

Zuordnung zum Curriculum:	Auflistung aller Studiengänge, in denen das Modul auftritt
Querweise:	Angabe, in welchem Zusammenhang das Modul zu anderen Modulen steht
SWS des Moduls:	Summe der SWS, die in allen Lehrveranstaltungen des Moduls anfallen
ECTS des Moduls:	Summe der ECTS-Punkte, die in allen Lehrveranstaltungen des Moduls erzielt werden können
Arbeitsaufwand:	Der Gesamtarbeitsaufwand in Stunden ergibt sich aus den ECTS-Punkten multipliziert mit 30 (Stunden). Der Zeitaufwand für das Eigenstudium ergibt sich, wenn vom Gesamtaufwand die Präsenzzeiten abgezogen werden. Diese ergeben sich wiederum aus den Semesterwochenstunden (SWS), die multipliziert mit 45 (Minuten) geteilt durch 60 die Präsenzzeit ergeben.
Voraussetzungen:	Module und Lehrveranstaltungen, die eine inhaltliche Grundlage für das jeweilige Modul darstellen. Bei Lehrveranstaltungen ist der Hinweis auf das jeweilige Modul enthalten, in dem die Lehrveranstaltung als Bestandteil auftritt.
Dauer:	Anzahl der Semester die benötigt werden, um das Modul abzuschließen
Häufigkeit:	Angabe, wie häufig ein Modul pro Studienjahr angeboten wird (jedes Semester bzw. jährlich)
Studien-/Prüfungsleistungen:	Auflistung aller Formen von Leistungsermittlung, die in den Veranstaltungen des Moduls auftreten
Sprache:	In der Regel werden die Lehrveranstaltungen aller Module auf Deutsch angeboten. Um Gaststudierenden unserer Partnerhochschulen, die nicht der deutschen Sprache mächtig sind, die Teilnahme an ausgewählten Lehrveranstaltungen zu ermöglichen, ist die Sprache in einigen Modulen als "deutsch/englisch" deklariert. Dieses wird den Partnerhochschulen mitgeteilt, damit sich die Interessenten für ihr Gastsemester entsprechende Veranstaltungen herausuchen können.
Lernziele des:	Übergeordnete Zielsetzungen hinsichtlich der durch das Modul zu vermittelnden Kompetenzen und Fähigkeiten aggregierter Form

Angaben zu den Lehrveranstaltungen

Lehrveranstaltung:	Bezeichnung der Lehrveranstaltung, die im Modul enthalten ist
Dozent(en):	Namen der Dozenten, die die Lehrveranstaltung durchführen
Hörtermin:	Angabe des Semesters, in dem die Veranstaltung nach Studienordnung gehört werden sollte
Art:	Angabe, ob es sich um eine Pflicht- oder Wahlveranstaltung handelt
Lehrform:	Lehrform kann Vorlesung, Praktikum, Seminar, u.v.m. sein
Semesterwochenstunden:	Eine Semesterwochenstunde dauert 70 Minuten und entspricht einer Vorlesungseinheit
ECTS:	Angabe der ECTS-Punkte, die in dieser Lehrveranstaltung des Moduls erzielt werden können
Medienformen:	Auflistung der Medienform(en), die in der Veranstaltung eingesetzt werden
Lernziele:	Stichwortartige Nennung die zentralen Lernziele der Lehrveranstaltung
Inhalt:	Gliederungsartige Auflistung der wesentlichen Inhalte der Lehrveranstaltung
Literatur:	Auflistung der wesentlichen Quellen, die den Studierenden zur Vertiefung zu den Veranstaltungsinhalten empfohlen werden. Es wird keine vollständige Auflistung aller Quellen gegeben, die als Grundlage für die Veranstaltung dienen.

I.1.2 Programmstrukturen 1

B003 Programmstrukturen 1

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B003
Bezeichnung	Programmstrukturen 1
Lehrveranstaltung(en)	B003a Programmstrukturen 1 B003b Übg. Programmstrukturen 1
Verantwortliche(r)	Prof. Dr. Andreas Häuslein
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul ist ein Einführungsmodul in den Themenbereich Programmierung für alle Studiengänge mit Informatikbezug. Die erworbenen Kompetenzen sind insbesondere die Grundlage für das Modul "Programmstrukturen 2", aber auch für die Module "Systemnahe Programmierung" und "UNIX und Shell-Programmierung".
Semesterwochenstunden	10
ECTS	5.0
Voraussetzungen	Grundkenntnisse der Schulmathematik, Basisfähigkeit zum abstrakten Denken.
Dauer	1

Lernziele

Nach Abschluss des Moduls verfügen die Studierenden über Grundkenntnisse hinsichtlich der Entwicklung von Programmen. Studierende mit Vorkenntnissen im Bereich der Programmierung sind in der Lage, diese fachlich fundiert einzuordnen und zu ergänzen.

Nach Abschluss des Moduls sind die unterschiedlichen Vorkenntnisse angeglichen und es ist eine gemeinsame Kompetenzbasis für die weiteren Veranstaltungen im thematischen Umfeld

der Programmierung und Software-Entwicklung gelegt.

Die Studierenden beherrschen sowohl die grundlegenden theoretischen Aspekte der Programmierung als auch die Basiskonzepte von imperativen, prozeduralen Programmiersprachen: Sie kennen alle wesentlichen Anweisungen zur Umsetzung algorithmischer Strukturen ebenso wie die typischen einfachen und strukturierten Datentypen. Dies schließt die Kenntnis einfacher dynamischer Datenstrukturen (dyn. Listen) hinsichtlich ihres Aufbaus und ihrer Verarbeitung mit ein. Die Studierenden kennen die Strukturierungsmöglichkeiten, die durch Prozeduren und Funktionen eröffnet werden und können diese zur Strukturierung ihrer Programme angemessen einsetzen. Die Studierenden können auf Basis dieser Kenntnis die programmiersprachlichen Mittel problemadäquat bei der Formulierung von Programmtexten nutzen.

Sie sind in der Lage, vollständige Programme begrenzter Komplexität eigenständig zu entwickeln und dabei die funktionale Korrektheit der Software sicherzustellen.

Die Studierenden kennen die typischen Funktionen einer Integrierten Entwicklungsumgebung und können diese angemessen zur Software-Entwicklung nutzen.

I.1.2.1 Programmstrukturen 1

Lehrveranstaltung	Programmstrukturen 1
Dozent(en)	Andreas Häuslein
Hörtermin	1
Häufigkeit	jedes Semester
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, E-Learning, Handout, interaktive Entwicklung und Diskussion von Modellen, Online-Aufbereitung, Overheadfolien, Softwaredemonstration, Tafel

Lernziele

Die Studierenden ...

- kennen die grundlegenden Konzepte imperativer Programmiersprachen und ihre Umsetzung in der Programmiersprache Pascal und können diese benennen.
- kennen die Syntax, Semantik und Pragmatik als wesentliche Aspekte von Programmiersprachen und können diese unterscheiden.
- kennen die wichtigsten Sprachbestandteile der Programmiersprache Pascal und beschreiben diese.
- setzen die Konzepte und Sprachbestandteile angemessen zur Lösung von Problemstellungen begrenzter Komplexität ein und bauen vollständige Programme für diese Problemstellungen auf.
- kennen die wesentlichen statischen Datenstrukturen imperativer Programmiersprachen, wählen bei der Programmierung zwischen diesen in Abhängigkeit von der Aufgabenstellung sicher aus und setzen sie angemessen zur Realisierung der Programmfunktionalität ein.
- kennen die Realisierung einfacher dynamischer Datenstrukturen und können diese zur Realisierung von Algorithmen nutzen.
- kennen wesentliche Qualitätskriterien für Software und können diese bei der Software-Entwicklung berücksichtigen.
- führen eine Fehlersuche und -beseitigung (Debugging) bei ihren Programmtexten durch.

Inhalt

- Grundkonzepte der Datenverarbeitung
- Entwurf und Darstellung von Algorithmen
- Allgemeine Aspekte von Programmiersprachen

- Daten in Programmen
 - Grundlegende Datentypen
 - Variablen, Zuweisungen, Konstanten
- Grundsätzlicher Aufbau von Programmen
- Operatoren und Ausdrücke
- Einfache und strukturierte Anweisungen
- Statische strukturierte Datentypen und ihre Nutzung
 - Strings
 - Arrays
 - Records
 - Sets
- Zeigertypen
 - Besonderheiten und Probleme bei der Nutzung von Zeigertypen
 - Aufbau dynamischer Datenstrukturen mit Hilfe von Zeigertypen
- Strukturierung von Programmen
 - Prozeduren und Funktionen
 - Units

Literatur

- OTTMANN, Thomas; WIDMAYER, Peter:
Programmierung mit PASCAL: Eine Einführung für Programmieranfänger, 9. Aufl., Springer Vieweg, 2018
- Collingbourne, Huw:
The Little Book Of Delphi Programming: Learn To Program with Object Pascal, Dark Neon, 2020
- CANTU, Marco:
Object Pascal Handbook, CreateSpace Independent Publishing Platform, 2015
- GUMM, Heinz-Peter; SOMMER, Manfred:
Einführung in die Informatik.
11. Aufl. München: Oldenbourg Wissenschaftsverlag, 2013.
- MATTHÄUS, Wolf-Gert:
Grundkurs Programmieren mit Delphi: Systematisch programmieren lernen für Einsteiger, 5. Aufl., Springer Vieweg, 2016
- WIRTH, Niklaus:
Algorithmen und Datenstrukturen: Pascal-Version. 5. Aufl., Teubner-Verlag, 2013

I.1.2.2 Übg. Programmstrukturen 1

Lehrveranstaltung	Übg. Programmstrukturen 1
Dozent(en)	Lars Neumann
Hörtermin	1
Häufigkeit	jedes Semester
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	6
ECTS	2.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Softwaredemonstration, studentische Arbeit am Rechner, Tafel, Tutorien

Lernziele

Die Studierenden ...

- festigen und vertiefen ihr Wissen zu den in der zugehörigen Vorlesung „Programmstrukturen 1“ vorgestellten Konzepten
- beherrschen die Arbeit mit einer modernen Entwicklungsumgebung (Embarcadero Delphi 10.3)
- lernen Grundlagen des Debugging und der Versionsverwaltung kennen
- erweitern ihre Teamfähigkeit durch die eigenständige praktische Anwendung des erlernten Wissens in Zweiergruppen

Inhalt

Ausgehend von den Grundlagen der Programmierung wie Datentypen, Verzweigungen und Iterationen werden in der Übung Programmstrukturen 1 in den einzelnen Aufgaben Ein- und Ausgabe, Operatoren, Bedingungen, Schleifen, Strings (sowohl über Stringfunktionen als auch über indizierten Zugriff), Arrays, Records, Mengen, Prozeduren und Funktionen, Zeiger und Listen sowie Dateien und Exceptions behandelt.

Die Inhalte höherer Aufgaben schließen dabei in der Regel die Inhalte der vorherigen mit ein.

Literatur

Skript:

- Vorlesungsskript unter <https://stud.fh-wedel.de/handout/Haeuslein/Programmstrukturen%201/>
- Weiteres Material unter <http://www.fh-wedel.de/mitarbeiter/ne/uebung-programmstrukturen-1/>

I.1.3 Informationstechnik

B004 Informationstechnik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B004
Bezeichnung	Informationstechnik
Lehrveranstaltung(en)	B004a Informationstechnik
Verantwortliche(r)	PD Dr. Dennis Säring
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) IT-Ingenieurwesen (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Das Modul "Informationstechnik" ist ein Einführungsmodul und soll ein breites Grundverständnis für die Funktionsweise von Rechnern vermitteln. Die erworbenen Kompetenzen stellen damit die Grundlagen für zum Beispiel die Module "Rechnerstrukturen und Digitaltechnik", "Systemsoftware" und "Großintegrierte Systeme" dar.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Keine
Dauer	1

Lernziele

Grundlegendes Verständnis für die Funktionsweise von Rechnern, sowohl aus Sicht der technischen Vorgänge und technischen Funktionselemente als auch aus informationstheoretischer Sicht.

Kenntnisse der rechnerinternen Abläufe auf allen technischen Beschreibungsebenen: vom Transistor, über Logikgatter und Schaltnetzen, hin zu Prozessorstrukturen, der Maschinenbefehlsebene und der Hochsprachenbefehlsebene.

Verständnis des quantitativen Informationsbegriffs und unterschiedlichen Kodierungsmöglichkeiten von Informationen, sowohl verlustfrei als auch verlustbehaftet.

Wissen um alternative Informationsverarbeitende Ansätze, die sich stark von der von-Neumann-Architektur unterscheiden.

I.1.3.1 Informationstechnik

Lehrveranstaltung	Informationstechnik
Dozent(en)	Dennis Säring
Hörtermin	1
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	5.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	E-Learning

Lernziele

Die Studierenden ...

- besitzen grundlegende Kompetenzen zum Verständnis der Funktionalität von Rechnern in Bezug auf ihre informationstheoretischen Grundlagen und deren praktische Implementierung
- können Vorgänge der Informationsverarbeitung auf der Maschinenebene theoretisch sowie praktisch umsetzen
- sind in der Lage die Umsetzung von Befehlen höherer Sprachebenen in Maschinenbefehle und in deren rechnerinternen Interpretation nachzuvollziehen
- kennen die Ansätze aktueller Rechnerstrukturen und Kommunikationsschnittstellen mit der Peripherie
- sind vertraut mit informationstheoretischen Ansätzen und unterschiedlichen Kodierungsverfahren.

Inhalt

- Grundlagen der Halbleitertechnik
- Logikgatter und Schaltnetze
- Zahlendarstellung und Berechnung
- FlipFlop und weitere Speicherstrukturen
- Moderne Rechnerarchitekturen
- Programmcode zu Assembler
- Computerperipherie
- Informationstheorie und Kodierung

Literatur

- Gumm, Hans-Peter; Sommer, Manfred: Einführung in die Informatik, Oldenbourg, 8. Auflage 2009.

- Müller, Käser, et., al. :Technische Informatik 1, vdf-Hochschulverlag Zürich, 2003
- Schiffmann, Schmitz: Technische Informatik 2, Grundlagen der Computertechnik, Springer-Verlag 1998
- Martin: Einführung in die Rechnerarchitektur, Fachbuchverlag Leipzig, 2003

I.1.4 Einführung in die Digitaltechnik

B006 Einführung in die Digitaltechnik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B006
Bezeichnung	Einführung in die Digitaltechnik
Lehrveranstaltung(en)	B006a Einführung in die Digitaltechnik B006b Prakt. Digitaltechnik
Verantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Informatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul "Einführung in Digitaltechnik" ist ein Einführungsmodul. Die erworbenen Kompetenzen stellen Grundlagen für zum Beispiel die Module "Rechnerstrukturen und Digitaltechnik", "Diskrete Systeme" und "Systementwurf mit VHDL" dar. Grundsätzlich kann das Modul sinnvoll mit den Modulen kombiniert werden, die ein Rechnersystem auf höheren Abstraktionsebenen (über dem Gatterniveau) behandeln. Das Modul ist fachübergreifend für alle Studiengänge relevant, die eine grundlegende Hardware-Kompetenz voraussetzen und hat somit eine direkte Verbindung zu den Studiengangszielen des Studiengangs "Bachelor Technische Informatik".
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Es werden mathematische Grundlagenkenntnisse entsprechend der Abitur-Stufe (überwiegend diskrete Mathematik) sowie das Verständnis einfacher technisch-naturwissenschaftlicher Zusammenhänge vorausgesetzt.
Dauer	1
Lernziele	Das Ziel des Moduls besteht in Vermittlung der allgemeinen Konzepte und Prinzipien des Aufbaus, der Funktionsweise sowie der Analyse und des Entwurfs digitaler Systeme. Angefangen bei Grundbegriffen (analoge und digitale Signale und Systeme, Schaltvariablen, Schaltalgebra usw.) lernen die Studierende Schaltnetze als technische Umsetzung von Schaltfunktionen

kennen. Als inhaltliche Voraussetzung für das Modul "Rechnerstrukturen und Digitaltechnik" werden schließlich die Speicherelemente diskutiert. Somit sind Studierende nach dem Abschluss des Moduls auf die Betrachtung komplexer, zustandsbehafteter Systeme vorbereitet, mit der das Modul "Rechnerstrukturen und Digitaltechnik" beginnt.

I.1.4.1 Einführung in die Digitaltechnik

Lehrveranstaltung	Einführung in die Digitaltechnik
Dozent(en)	Sergei Sawitzki
Hörtermin	1
Häufigkeit	jedes Semester
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Tafel

Lernziele

Die Studierenden ...

- gewinnen eine Einsicht in die mathematischen Grundlagen der digitalen Schaltungstechnik
- erlernen die Methoden zur Analyse, Darstellung und Vereinfachung von Schaltfunktionen
- begreifen ein Schaltnetz als technische Umsetzung einer Schaltfunktion
- lernen die wichtigsten Grundelemente digitaler Systeme kennen
- erwerben die Fähigkeit, einfache digitale Systeme zu begreifen, zu spezifizieren, zu entwerfen und zu optimieren

Inhalt

- Einleitung: Digitale Systeme
- Mathematische Grundlagen
 - Entstehungsgeschichte
 - Aussagenlogik und Boolesche Algebra
 - Schaltalgebra, Schaltfunktionen und Schaltfunktionssysteme
 - Operatorensysteme
 - Normalformen und Dualitätsprinzip
- Schaltnetze
 - Darstellung
 - Vereinfachung (KV-Diagramme, QMCV, BDDs)
 - Analyse (Funktion, Komplexität, Zeitverhalten)
 - Synthese und Realisierung
 - Beispiele

- Speicherelemente

Literatur

- Hoffmann, Dirk: Grundlagen der technischen Informatik, Carl Hanser Verlag 2007
- Schiffmann, Wolram; Schmitz, Robert: Technische Informatik, in 3 Bänden. 3. Auflage Springer Verlag, 1996
- Beuth, Klaus: Elektronik 4. Digitaltechnik, 13. Auflage Vogel Verlag und Druck 2003

I.1.4.2 Prakt. Digitaltechnik

Lehrveranstaltung	Prakt. Digitaltechnik
Dozent(en)	Dieter Opitz
Hörtermin	1
Häufigkeit	jedes Semester
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Praktikumsbericht / Protokoll
Sprache	deutsch
Lehr- und Medienform(en)	

Lernziele

Fähigkeit zur Übertragung des theoretischen Wissens in eine erlebbare, reale technische Umgebung.

Inhalt

- Schaltnetzentwurf Schaltnetz wird aus einer gegebenen Spezifikation formal entworfen. Der Entwurf wird auf einem IC-Trainer realisiert. Die Schaltung wird auf Funktion und Einhaltung der Spezifikation überprüft. Die Ergebnisse werden dokumentiert.
- Schaltwerkentwurf Schaltwerk (z. B. ein Zähler) wird aus einer gegebenen Spezifikation formal entworfen. Der Entwurf wird auf einem IC-Trainer realisiert. Die Schaltung wird auf Funktion und Einhaltung der Spezifikation überprüft. Die Ergebnisse werden dokumentiert.

Literatur

Vorlesungsskript

I.1.5 Analysis

B001 Analysis

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B001
Bezeichnung	Analysis
Lehrveranstaltung(en)	B001a Analysis B001b Übg. Analysis
Verantwortliche(r)	Dr. Hendrik Glowatzki
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Das Modul ist sinnvoll mit anderen Modulen der Mathematik zu kombinieren und zur Bildung mathematischer Grundlagenkompetenzen in allen naturwissenschaftlichen, ingenieurtechnischen und wirtschaftswissenschaftlichen Studiengängen verwendbar. Es stellt Querbezüge zur Finanzmathematik, Linearen Algebra, Statistik, Physik und Betriebswirtschaftslehre her.
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Voraussetzung zur Teilnahme am Modul sind schulische Grundlagen der Mathematik. Insbesondere gehören hierzu die grundlegenden Begriffe über Mengen, das Rechnen mit reellen Zahlen, Gleichungen mit einer Unbekannten, Basiswissen zur elementaren Geometrie sowie zu Funktionen und Kurven.
Dauer	1

Lernziele

Nachdem erfolgreichen Abschluss des Moduls besitzen die Studierenden Rechenfertigkeiten, anschauliche Vorstellungen und theoretisches Verständnis von Funktionen. Sie können dieses auf Funktionen einer reellen Veränderlichen anwenden, Problemstellungen und Lösungswege klassifizieren und bewerten sowie Problemlösungen prüfen und beurteilen. Sie beherrschen die Grundbegriffe der Analysis einer Veränderlichen, können dieses auf Funktionen mehrere Veränderlicher übertragen und als Fundament für die weiteren fachwissenschaftlichen Studien nutzen. Sie verfügen über formalisierte mathematische Denk- und Arbeitsweisen und sind befähigt mathematische Kausalzusammenhänge aufzustellen und sich in neue formale Systeme einzuarbeiten. Die Studierenden erkennen die Querbezüge der Analysis zu anderen mathematischen und fachspezifischen Fächern

Durch die Übungen erarbeiten sie sich einen sicheren, präzisen und selbständigen Umgang mit den in den Vorlesungen behandelten Begriffen, Aussagen und Methoden. Praxisorientierte Problemstellungen können sie in mathematische Beziehungen und Modelle umsetzen und anhand dieser Modelle bearbeiten. Sie können die Praxisrelevanz der Analysis für verschiedene Fachgebieten bewerten und die Analysis auf Problemstellungen aus Informatik, Technik und Ökonomie anwenden.

I.1.5.1 Analysis

Lehrveranstaltung	Analysis
Dozent(en)	Hendrik Glowatzki
Hörtermin	1
Häufigkeit	jedes Semester
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Handout, Tutorien

Lernziele

Die Studierenden ...

- kennen und verstehen die grundlegenden Begriffe, Aussagen und Methoden der Analysis,
- können mathematische Regeln korrekt anwenden,
- verstehen Beweistechniken,
- erkennen die fundamentale Bedeutung des Grenzwertbegriffes für die Analysis,
- beherrschen die Methoden des Differenzierens und Integrierens,
- können die eindimensionale Differentialrechnung bei praxisorientierten Fragestellungen flexibel in unterschiedlichen Fachgebieten einsetzen und dabei beurteilen, welche analytischen Hilfsmittel für welche Problemstellungen zielführend sind,
- erkennen die Anwendbarkeit und den Nutzen der Analysis für unterschiedliche Fachgebiete und deren spezifischen Problemstellungen,
- können praxisorientierte Problemstellungen in mathematische Beziehungen bzw. Modelle umzusetzen und anhand analytischer Modelle weiter bearbeiten
- können neue, unklare und ungewöhnliche Aufgabenstellungen als solche erkennen und zur Bearbeitung weiterführende Hilfestellung in Anspruch nehmen,
- verfügen über gesteigerte Kompetenzen sich Fähigkeit durch Selbststudium anzueignen und sich in neue formale Systeme einzuarbeiten

Inhalt

- Zahlentypen
- Folgen
 - Bildungsgesetze
 - Grenzwerte
- Funktionen, Relationen
 - Funktionstypen

- Umkehrfunktion
- Differentialrechnung
 - Differentiationsregeln
 - Anwendungen der Differentialrechnung (Kurvendiskussionen und Extremwerte)
- Integralrechnung
 - Integrationsmethoden
 - Anwendungen der Integralrechnung (Bestimmte Integrale)
- Funktionen mit zwei Variablen
 - Partielle Differentiation
 - Extremwertaufgaben mit Nebenbedingungen

Literatur

- BÖHME, Gert:
Analysis 1.
6. Aufl. Berlin: Springer-Verlag, 1990
- FETZER, Albert; FRÄNKEL, Heiner:
Mathematik 1.
10. bearbeitete Aufl. Berlin: Springer-Verlag, 2008
- FETZER, Albert; FRÄNKEL, Heiner:
Mathematik 2.
6. korrigierte Aufl.. Berlin: Springer-Verlag, 2009
- HENZE, Norbert; Last, Günter:
Mathematik für Wirtschaftsingenieure 1.
2. Aufl. Wiesbaden: Vieweg + Teubner Verlag, 2005
- KUSCH, Lothar:
Mathematik. Aufgabensammlung mit Lösungen. Bd. 3
9. Aufl. Berlin: Cornelsen Verlag, 1995
- OHSE, Dietrich: Mathematik für Wirtschaftswissenschaftler 1. Analysis.
6. Aufl. München: Verlag Vahlen, 2004
- PAPULA, Lothar:
Mathematik für Ingenieure und Naturwissenschaftler 1: Ein Lehr- und Arbeitsbuch für
das Grundstudium.
12. überarbeitete und erweiterte Aufl. Wiesbaden: Vieweg + Teubner Verlag, 2009
- PREUSS, Wolfgang; WENISCH, Günter:
Lehr- und Übungsbuch Mathematik 1: Grundlagen - Funktionen - Trigonometrie.
2. neu bearbeitete Aufl. München: Carl Hanser Verlag, 2003
- PREUSS, Wolfgang; WENISCH, Günter:
Lehr- und Übungsbuch Mathematik 2: Analysis.
3. Aufl. München: Carl Hanser Verlag, 2003

I.1.5.2 Übg. Analysis

Lehrveranstaltung	Übg. Analysis
Dozent(en)	Fikret Koyuncu
Hörtermin	1
Häufigkeit	jedes Semester
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Teilnahme
Sprache	deutsch
Lehr- und Medienform(en)	Handout, Tafel

Lernziele

Die Studierenden können ...

- praktische Problemstellungen mathematisch formulieren
- beurteilen, welche analytischen Hilfsmittel zielführend sind
- neue, unklare und ungewöhnliche Aufgabenstellungen als solche erkennen und mit weiterführender Hilfestellung bearbeiten
- Lösungsansätze präsentieren und begründen

Inhalt

- Bearbeitung von Übungsaufgaben aus dem Themenspektrum der zugehörigen Lehrveranstaltung
- Vorstellung und Diskussion möglicher Lösungswege
- Heranführung an mathematische Softwaretools

Literatur

PAPULA, Lothar: Mathematik für Ingenieure und Naturwissenschaftler: Klausur- und Übungsaufgaben 4. überarbeitete und erweiterte Aufl. Wiesbaden: Vieweg + Teubner Verlag, 2010

I.1.6 Mathematische Konzepte und Diskrete Mathematik

B002 Mathematische Konzepte und Diskrete Mathematik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B002
Bezeichnung	Mathematische Konzepte und Diskrete Mathematik
Lehrveranstaltung(en)	B002a Diskrete Mathematik
Verantwortliche(r)	Prof. Dr. Sebastian Iwanowski
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul ist ein Einführungsmodul. Es liefert die Konzepte für ein tieferes Verständnis der anderen Mathematikmodule wie "Analysis" und "Lineare Algebra". Die vermittelten Konzepte und Inhalte werden gebraucht in den Modulen "Informationstechnik", "Einführung in Digitaltechnik", "Programmstrukturen 1 und 2", "Formale Sprachen", "Algorithmen und Datenstrukturen", "Einführung in Datenbanken" und "Anwendungen der Künstlichen Intelligenz". Außerdem werden die in diesem Modul vermittelten Kenntnisse in allen Mastervorlesungen der IT-orientierten Studiengänge vorausgesetzt.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Die Studierenden müssen auf dem Kenntnisstand der Schulmathematik der 9. Klasse (Gymnasium) sein. Sie sollten insbesondere mit den Mengen der natürlichen, ganzen, rationalen und reellen Zahlen sowie mit den dafür geltenden Rechengesetzen vertraut sein. Außerdem wird ein gutes logisches Denkvermögen vorausgesetzt.

Dauer

1

Lernziele

Nach Abschluss de Moduls verstehen und beherrschen die Studierenden allgemeine formalisierte mathematische Denk- und Arbeitsweisen. Sie kennen grundlegende Beweistechniken und verstehen die Notwendigkeit mathematischen Beweisens. Ferner verfügen sie über die Fähigkeit, Kausalzusammenhänge nachzuvollziehen und zu erarbeiten. Sie können mathematische Regeln korrekt anwenden. Die Studierenden sind in der Lage, die Verwendbarkeit der vermittelten mathematischen Hilfsmittel auf praktische Problemstellungen kompetent zu beurteilen. Sie können praxisorientierte Problemstellungen in mathematische Beziehungen bzw. Modelle umsetzen und anhand dieser Modelle bearbeiten und lösen. Ferner können sie sich in neue formale Systeme einarbeiten und dessen Regelwerke richtig anwenden. Schließlich besitzen sie die Fähigkeit, neue, unklare und ungewöhnliche Aufgabenstellungen als solche zu erkennen und zu ihrer Bearbeitung weiterführende Hilfestellung in Anspruch zu nehmen. Im Speziellen beherrschen sie die wesentlichen Konzepte der Diskreten Mathematik und können diese auf anwendungsbezogene Problemstellungen in den Gebieten der Informatik, Technik und Wirtschaft anwenden.

I.1.6.1 Diskrete Mathematik

Lehrveranstaltung	Diskrete Mathematik
Dozent(en)	Sebastian Iwanowski
Hörtermin	1
Häufigkeit	jedes Semester
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	5.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Overheadfolien, Softwaredemonstration, Tafel, Tutorien

Lernziele

Nach Abschluss der Veranstaltung besitzen die Studierenden folgende Kompetenzen:

- Beherrschen der grundlegenden mathematischen Begriffe und Konzepte (Definition, Satz, Beweis) und Fähigkeit zur Unterscheidung derselben.
- Beherrschen der Grundlagen und der Formalisierung logischen Denkens.
- Verständnis elementarer Logik und Mengenlehre und des inneren Zusammenhangs dieser Gebiete.
- Darauf aufbauendes Verständnis von Relationen und Funktionen.
- Fähigkeit, elementare Beweisprinzipien wie vollständige Induktion in verschiedenen Kontexten anzuwenden.
- Beherrschen der grundlegenden Sätze der elementaren Zahlentheorie, Gruppen- und Körpertheorie, Kombinatorik und Graphentheorie und selbständige Anwendung an Beispielen.

Inhalt

- Logik
 - Einführung
 - Aussagenlogik
 - Prädikatenlogik
- Mengenlehre
 - Grundlegende Begriffe und Konzepte
 - Relationen
 - Funktionen
 - Boolesche Algebren
- Beweisführung

- Strukturen der mathematischen Beweisführung
- Vollständige Induktion
- Beweisstrategien
- Zahlentheorie
 - Teilbarkeit
 - Teilen mit Rest
 - Primzahlen
 - Modulare Arithmetik
- Algebraische Strukturen
 - Gruppen
 - Körper
- Kombinatorik
 - Zählformeln für Mengen
 - Permutationen
- Graphentheorie
 - Terminologie und Repräsentation
 - Wege in Graphen
 - Bäume
 - Planare Graphen
 - Färbungen

Literatur

- Sebastian Iwanowski / Rainer Lang:
Diskrete Mathematik mit Grundlagen, Springer 2014, ISBN 978-3-658-07130-1 (Print),
978-3-658-07131-8 (Online)
- Albrecht Beutelspacher / Marc-Alexander Zschiegner:
Diskrete Mathematik für Einsteiger.
Vieweg 2004 (2. Auflage), ISBN 3-528-16989-3
- Norman L. Biggs:
Discrete Mathematics.
Oxford University Press 2002, ISBN 0-19-850717-8
- Neville Dean: Diskrete Mathematik.
Pearson Studium, Reihe "im Klartext" 2003, ISBN 3-8273-7069-8
- Christoph Meinel / Martin Mundhenk:
Mathematische Grundlagen der Informatik.
Teubner 2002 (2. Auflage), ISBN 3-519-12949-3

I.1.7 Mechanik und Elektrotechnik

B252 Mechanik und Elektrotechnik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B252
Bezeichnung	Mechanik und Elektrotechnik
Lehrveranstaltung(en)	B252a Grundlagen der Mechanik B252a Grundlagen der Elektrotechnik
Verantwortliche(r)	Prof. Dr.-Ing. Carsten Burmeister
Zuordnung zum Curriculum	Data Science & Artificial Intelligence (Bachelor) IT-Ingenieurwesen (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Das Modul bereitet auf weiterführende Fächer der Ingenieurwissenschaften und technischen Informatik vor. So ist es z.B. mit der Übertragungstechnik zu kombinieren oder mit Industrie 4.0.
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Die Studierenden sollen Schulkenntnisse der Physik und der Mathematik besitzen.
Dauer	1

Lernziele

Nach Abschluss des Moduls verfügen die Studierenden über Kenntnisse grundlegender physikalischer Gesetzmäßigkeiten. Dadurch besitzen sie ein Verständnis der Mechanik, der Elektrizitätslehre und der Elektronik in einer dem Studiengang angemessenen Tiefe. Dazu zählen die Kenntnisse der physikalischen Grundlagen von Kräften und Bewegungen. Die Studierenden verfügen über Kenntnisse hinsichtlich der Berechnung linearer Gleichstromkreise. Dadurch sind sie in der Lage, lineare elektrische Kreise, z.B. in der Energie- oder Nachrichtenübertragung zu errechnen. Ferner besitzen sie die Fähigkeit zur Abstraktion bei der Beschreibung komplexer linearer Systeme, speziell Matrixgleichungssysteme. Außerdem haben Sie grundlegende Kenntnisse elektrischer und magnetischer Felder, deren Berechnungsmethoden und Anwendungen in Form von elektrischen Bauelementen wie der Spule und dem Kondensator.

I.1.7.1 Grundlagen der Mechanik

Lehrveranstaltung	Grundlagen der Mechanik
Dozent(en)	Andreas Haase
Hörtermin	1
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	E-Learning, Online-Aufbereitung

Lernziele

Die Lernenden beherrschen nach erfolgreichem Besuch der Lehrveranstaltung grundlegende physikalischen Gesetzmäßigkeiten und verstehen die Arbeitsweise der Physik, die zum Verständnis mechanischer, aber auch in nachfolgenden Veranstaltungen zu behandelnde nicht-mechanischer Phänomene erforderlich sind. Sie können ...

- die vorgestellten physikalischen Begriffe und Gesetze der Mechanik selbständig erklären und zueinander in Beziehung setzen bzw. gegeneinander abgrenzen.
- für ausgesuchte Aufgaben aus der Mechanik selbständig eine Lösungsstrategie entwickeln.
- Aufgaben unter Anwendung der erlernten physikalischen und mathematischen Mittel und Methoden eigenständig lösen.
- das Ergebnis einer gelösten Aufgabe kritisch bewerten und daraus Schlüsse und Folgerungen ziehen.

Inhalt

- Maßsystem und Einheiten
- Kinematik
- Dynamik (Translation und Rotation)
- Die Newtonschen Gesetze
- Arbeit, Leistung und Energie
- Impuls- und Energieerhaltung
- Reibungskräfte
- Bewegung starrer Körper

Literatur

Halliday, Resnick, Walker: Physik, Bachelor Edition, Wiley-VCH (2013)

Kersten (Hrsg.), Tipler: Physik für Studierende der Naturwissenschaften und Technik,
Springer Spektrum (2019)

Meschede: Gerthsen Physik, Springer Spektrum (2015)

I.1.7.2 Grundlagen der Elektrotechnik

Lehrveranstaltung	Grundlagen der Elektrotechnik
Dozent(en)	Carsten Burmeister
Hörtermin	1
Häufigkeit	jährlich
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assigm.
Semesterwochenstunden	4
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch/englisch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, studentische Arbeit am Rechner, Tafel

Lernziele

Die Studierenden ...

- besitzen ein Verständnis linearer elektrotechnischer Grundzusammenhänge und deren Wirkungsweisen in Gleichstromkreisen.
- haben Kenntnis der Anwendung von linearen elektrischen Kreisen in der Energieübertragung, in der Nachrichtenübertragung und bei Übergangsvorgängen.
- haben die Fähigkeit, Wirkungsweisen linearer Schaltungen zu verstehen und zu berechnen.
- besitzen die Fähigkeit zur Abstraktion bei der Beschreibung komplexer linearer Systeme, speziell Matrixgleichungssysteme.

Inhalt

- Physikalische Größen, Einheiten, Gleichungen
- Lineare Gleichstromkreise
 - Grundbegriffe: Strom, Spannung, Arbeit, Leistung, Wirkungsgrad
 - Das Ohmsche Gesetz
 - Spannungsquellen
 - Stromquellen
 - Die Kirchhoffschen Sätze
 - Strom- und Spannungsteiler
 - Berechnung von Netzwerken mit einer Quelle
 - Lineare Überlagerung mehrerer Quellen
 - Ersatzspannungs- und -stromquellen
 - Leistungsanpassung

- Knotenpotenzialverfahren
- Das Kondensatorgesetz
 - Elektrische Ladung und ihre Wirkung
 - Kapazität von Kondensatoren
 - Energie des elektrischen Feldes
 - Zusammenschaltung von Kondensatoren
- Das Induktionsgesetz
 - Magnetische Feldgrößen
 - Durchflutungsgesetz
 - Ferromagnetismus
 - Induktion
 - Energie des magnetischen Feldes
 - Selbst- und Gegeninduktivität
 -

Literatur

- Hagmann, G.: Grundlagen der Elektrotechnik. Aula-Verlag, 2000 (7. Auflage)
- Führer, A.; Heidemann, K.; Nerretter, W.: Grundgebiete der Elektrotechnik, Bd. 1: Stationäre Vorgänge. Hanser-Verlag, 1990
- Paul, R.: Elektrotechnik: Grundlagenlehrbuch, Bd. 1: Felder und einfache Stromkreise. Springer-Verlag, 1993 (3. Auflage)
- Paul, S.: Grundlagen der Elektrotechnik und Elektronik 1: Gleichstromnetzwerke und ihre Anwendungen. Springer-Verlag, 2014 (5. Auflage)
- Papula, L.: Mathematik für Ingenieure, Bd. 2. Vieweg, 2000 (9. Auflage)

I.1.8 Programmstrukturen 2

B020 Programmstrukturen 2

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B020
Bezeichnung	Programmstrukturen 2
Lehrveranstaltung(en)	B020a Programmstrukturen 2 B020b Übg. Programmstrukturen 2
Verantwortliche(r)	Prof. Dr. Andreas Häuslein
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul basiert auf den im Modul "Programmstrukturen 1" erworbenen Kompetenzen. Es schafft die Grundlagen für Module der fortgeschrittenen Programmierung in Informatik-Studiengängen, zum Beispiel die Module "Algorithmen und Datenstrukturen", "Fortgeschrittene Objekt-orientierte Programmierung" und "Web-Anwendungen".
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Kenntnis der Grundkonzepte imperativer Programmiersprachen hinsichtlich der wesentlichen statischen und einfachen dynamischen Datenstrukturen sowie der Anweisungen zur Umsetzung der algorithmischen Grundstrukturen, Fähigkeit zur Erstellung von vollständigen Programmen begrenzter Komplexität. Die erfolgreiche Teilnahme an der Übung "B003b Übg. Programmstrukturen 1" ist Voraussetzung, um an der Übung "B020b Übg. Programmstrukturen 2" teilzunehmen.
Dauer	1

Lernziele

Nach Abschluss des Moduls kennen die Studierenden die Basiskonzepte der Objektorientierten Programmierung und können diese Kenntnisse zur Erstellung von objektorientierter Software begrenzten Umfangs einsetzen. Die Studierenden wissen, wie die Programmiersprache Java grundsätzlich aufgebaut ist, sie kennen die grundlegenden Sprachelemente der Programmiersprache Java und können diese sicher zur Realisierung entsprechender algorithmischer Strukturen nutzen.

Die Studierenden kennen die zentralen Konzepte der Objektorientierten Programmierung (z.B. Vererbung und Dynamische Bindung) und können sie zur Realisierung objektorientierter Software angemessen einsetzen. Die Studierenden können Bezüge zwischen der imperativ prozeduralen Sprache Pascal und der objektorientierten Programmiersprache Java herstellen. Sie sind damit in der Lage, wesentliche allgemeine Konzepte von Programmiersprachen zu erkennen und einzuordnen.

Die Studierenden besitzen die Fähigkeit, eine moderne Entwicklungsumgebung zur Software-Erstellung zu nutzen. Sie können mit den Mitteln der objektorientierten Sprache Java einfache rekursive Datenstrukturen (Listen) aufbauen, kennen grundlegende Algorithmen für diese Datenstrukturen und können Variationen dieser Algorithmen eigenständig entwickeln.

Die Studierenden sind in der Lage die Modularisierungskonzepte der Sprache Java, soweit sie Gegenstand der Vorlesung sind, zu einer problemadäquaten Strukturierung eines Programms mittleren Umfangs und begrenzter Funktionalität einzusetzen.

Sie kennen bezogen auf die Gestaltung einer grafischen Benutzungsoberfläche die wesentlichen Regeln und Richtlinien und sind in der Lage diese für die Gestaltung konkreter Oberflächen einzusetzen. Sie besitzen die Kenntnis hinsichtlich einer konkreten technischen Umsetzung von grafischen Oberflächen und können diese zur Implementation solcher Oberflächen nutzen.

Sie verfügen über Basiskenntnisse hinsichtlich der Qualitätssicherung von Software in Form einfacher Testverfahren und können diese einsetzen, um die funktionale Korrektheit und ein ausreichendes Maß an Zuverlässigkeit der Software zu gewährleisten.

I.1.8.1 Programmstrukturen 2

Lehrveranstaltung	Programmstrukturen 2
Dozent(en)	Andreas Häuslein
Hörtermin	2
Häufigkeit	jedes Semester
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, E-Learning, Handout, Online-Aufbereitung, Softwaredemonstration, Tafel

Lernziele

Die Studierenden ...

- identifizieren die Basiskonzepte der Objektorientierten Programmierung und stellen diese den Konzepten der prozeduralen Programmierung gegenüber.
- entwickeln Software auf der Grundlage der Kernkonzepte der Objektorientierten Programmierung.
- stellen die grundlegenden Sprachelemente (Datentypen, Anweisungen, Realisierung von objektorientierten Konzepten) von Java zusammen und wählen daraus aus, um Java-Programme mittlerer Komplexität zu entwickeln.
- vergleichen die Programmiersprachen Pascal und Java und stellen ihre Gemeinsamkeiten und Unterschiede heraus.
- setzen eine moderne Entwicklungsumgebung zur Unterstützung der Softwareentwicklung ein und stellen die damit verbundenen Funktionalitäten und Vorgehensweisen dar.
- entwerfen einfache dynamische Datenstrukturen im Kontext einer objektorientierten Programmiersprache.
- erläutern grundlegende Algorithmen, die auf den vermittelten Datenstrukturen arbeiten.
- entwerfen für Programme mittlerer Komplexität durch Einsatz geeigneter Elemente der Programmiersprache Java eine angemessene Modularisierung und legen entsprechende Schnittstellen zwischen den Modulen fest.
- benennen die Grundregeln der benutzungsgerechten Gestaltung von Programmen und nutzen diese, um Benutzungsoberflächen von Programmen begrenzter Funktionalität sowohl strukturell als auch funktional angemessen zu gestalten.
- kennen die grundlegenden Klassen und ihre Operationen, mit denen dateibezogene Operationen implementiert werden können.

Inhalt

- Grundkonzept der Programmiersprache Java

- Grundlegende Eigenschaften der Sprache
- Grundlegender Aufbau von Java-Programmen
- Ausführung von Java-Programmen
- Vorstellung der eingesetzten Entwicklungsumgebung (NetBeans)
- Grundlegende Programmelemente
 - Primitive Datentypen in Java
 - Variablen, Zuweisung, Gültigkeitsbereiche
 - Operatoren und Ausdrücke
 - Anweisungen
- Referenzdatentypen
 - Arrays
 - Klassen
- Statische Methoden
- Grundlegende Klassen
 - String
 - StringBuilder
 - Wrapper-Klassen für primitive Datentypen
 - Enum
- Grundkonzepte der Objektorientierung
 - Klassen und Instanzen mit Attributen und Methoden
 - Sichtbarkeit, Packages
 - Konstruktoren
 - Vererbung und Überschreiben
 - Dynamisches Binden, Polymorphie
 - Objektorientierte Realisierung rekursiver dynamischer Datenstrukturen (Listen)
 - Generische Typen
 - Abstrakte Klassen und Interfaces - Deklaration und Nutzung
 - Realisierung grafischer Benutzungsoberflächen
 - Behandlung von Laufzeitfehlern
 - Klassen zur Realisierung von Dateioperationen

Literatur

- ULLENBOOM, Christian:
Java ist auch eine Insel. 15. Auflage, Rheinwerk Verlag, 2020
- HABELITZ, Hans-Peter:
Programmieren lernen mit Java. 6. Auflage, Rheinwerk Computing, 2020
- SOLYMOSI, Andreas, GRUDE, Ulrich:
Grundkurs Algorithmen und Datenstrukturen in JAVA: Eine Einführung in die praktische Informatik. 6. Auflage, Springer Vieweg, 2017
- RATZ, Dietmar:
Grundkurs Programmieren in Java. 7. Auflage,
Hanser Verlag, 2014
- ABTS, Dietmar:
Grundkurs Java, Von den Grundlagen bis zu Datenbank- und Netzanwendungen, Springer Vieweg, 2015
- STEYER, Ralph:
Einführung in JavaFX: Moderne GUIs für RIAs und Java-Applikationen.
Springer-Vieweg, 2014
- EPPLE, Anton:
JavaFX 8: Grundlagen und fortgeschrittene Techniken. dpunkt.verlag, 2015
- SALTER, David; DANTAS, Rhawi:
NetBeans IDE 8 Cookbook. Packt Publishing, 2014

I.1.8.2 Übg. Programmstrukturen 2

Lehrveranstaltung	Übg. Programmstrukturen 2
Dozent(en)	Gerit Kaleck
Hörtermin	2
Häufigkeit	jedes Semester
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, studentische Arbeit am Rechner

Lernziele

Die Studierenden erwerben in der Übung die ...

- Fähigkeit zum praktischen Einsatz der Basiskonzepte objektorientierter Programmiersprachen sowie ihrer Umsetzung in Java.
- Fähigkeit zum Aufbau einfacher dynamischer Datenstrukturen im Kontext einer objektorientierten Programmiersprache und Fähigkeit der Anwendung grundlegender Algorithmen auf diesen Datenstrukturen.
- Fähigkeit zur Nutzung einer aktuellen, verbreiteten Entwicklungsumgebung.
- Fähigkeit zur Realisierung von vollständigen Software-Systemen kleineren Umfangs ausgehend von einer verbalen Aufgabenstellung.
- Fähigkeit zur Software-Entwicklung im kleinen Team.
- Fähigkeit zur Ermittlung geeigneter Testfälle zur Qualitätssicherung.
- Kenntnis der Grundregeln zur Gestaltung benutzungsgerechter Oberflächen und bedienfreundlicher Software.

Inhalt

- Einführung in die Programmierung mit Java und die Entwicklungsumgebung.
- Anwendung der in der Vorlesung vorgestellten Grundkonzepte der objektorientierten Programmierung durch das Lösen verbal formulierter Aufgabenstellungen in kleinen Teams.
- Testen und Präsentieren der sauber strukturierten Lösung.

Literatur

- HABELITZ, Hans-Peter: Programmieren lernen mit Java: Der leichte Einstieg für Programmieranfänger, Rheinwerk Computing, 2017 (5. Auflage) ISBN-13: 978-3836256056
- KOFLER, Michael: Java: Der kompakte Grundkurs mit Aufgaben und Lösungen im Taschenbuchformat, Rheinwerk Computing, 2019 (3. Auflage) ISBN-13: 978-3836269582

- ULLENBOOM, Christian: Java ist auch eine Insel, Rheinwerk Computing, 2020 (15. Auflage) ISBN-13: 978-3836277372

I.1.9 Rechnerstrukturen und Digitaltechnik

B023 Rechnerstrukturen und Digitaltechnik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B023
Bezeichnung	Rechnerstrukturen und Digitaltechnik
Lehrveranstaltung(en)	B023a Digitaltechnik 2 B023a Rechnerstrukturen
Verantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Informatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul "Rechnerstrukturen und Digitaltechnik" baut auf den im Modul "Einführung in Digitaltechnik" erworbenen Kenntnissen und Fähigkeiten auf. Die im Modul "Rechnerstrukturen und Digitaltechnik" erworbenen Kompetenzen stellen die Grundlagen für zum Beispiel die Module "Diskrete Systeme" und "Systementwurf mit VHDL" dar. Das Modul kann sinnvoll mit den Modulen, die einerseits Grundlagen der Digitaltechnik beleuchten und andererseits ein Rechnersystem auf höheren Abstraktionsebenen (über dem Gatterniveau) behandeln, kombiniert werden. Das Modul ist fachübergreifend für alle Studiengänge relevant, die eine grundlegende Hardware-Kompetenz sowie Kenntnisse moderner Rechnerarchitekturen voraussetzen und hat somit eine direkte Verbindung zu den Studiengangszielen des Studiengangs "Bachelor Technische Informatik".
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Es werden Kompetenzen aus dem Modul "Einführung in Digitaltechnik" vorausgesetzt.
Dauer	1

Lernziele

Aufbauend auf den im Modul "Einführung in Digitaltechnik" erworbenen Kompetenzen bekommen die Studierenden das Verständnis für zustandsbehaftete Systeme und deren Implementierung in Form von Schaltwerken vermittelt. Die Abstraktion mathematischer Darstellung

von Zustandsautomaten ist nach dem Absolvieren des Moduls mit ihren technischen Abbildern versehen. Dabei lernen die Studierenden zu begreifen, worin die Unterschiede zwischen Modellen und realen Schaltungen und Systemen bestehen, warum Abstraktionen und modellhafte Darstellungen unvermeidlich sind und wo deren Grenzen liegen. Aufbauend auf den einfacheren Schaltungen werden Rechnersysteme als komplexe Vertreter digitaler Systeme betrachtet. Die Studierende lernen den Aufbau und die Funktionsweise moderner Rechner kennen und mit den Begriffen und Konzepten aus dem Bereich Rechnerarchitektur sicher umzugehen. Sie sollten in die Lage versetzt werden, Abläufe in Hardware eines modernen Rechners zu begreifen und klassische sowie innovative Architekturkonzepte zu erkennen und richtig einzuordnen. Ein wesentliches Lernziel besteht außerdem in Erkennung der Bedeutung der Zeitverhaltens von einfachen logischen Gattern und Schaltungen und deren Einflusses auf die Leistungsfähigkeit digitaler Systeme (Verzögerungs-, Setz und Haltezeiten, Taktfrequenz, Steigerung des Durchsatzes in modernen Rechnerarchitekturen).

I.1.9.1 Digitaltechnik 2

Lehrveranstaltung	Digitaltechnik 2
Dozent(en)	Sergei Sawitzki
Hörtermin	2
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.5
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Tafel

Lernziele

Die Studierenden ...

- ergänzen und erweitern die Grundlagen aus der Lehrveranstaltung Digitaltechnik 1
- erlangen Kenntnisse über Aufbau und Funktionsweise von Speicherelementen und Schaltwerken
- erlernen die Methoden zur Analyse, Darstellung und Vereinfachung von endlichen Zustandsautomaten
- begreifen ein Schaltwerk als technische Umsetzung eines endlichen Zustandsautomaten
- gewinnen eine Einsicht in die Methoden der Zeitverhaltensanalyse und Zeitverhaltensoptimierung von digitalen Systemen, sowie lernen das Zeitverhalten und die Zeitvorgaben beim Entwurf digitaler Systemen zu berücksichtigen;
- erlangen die Fähigkeit, digitale Systeme in der Gesamtheit verschiedener Aspekte zu begreifen, die für ihren praktischen Einsatz eine Rolle spielen (Schnittstellen, Komplexität, Zeitverhalten, Leistungsaufnahme, usw.)
- erlangen die Fähigkeit, digitale Systeme mittlerer Komplexität zu begreifen, zu spezifizieren, zu entwerfen und zu optimieren.

Inhalt

- Einleitung
 - Lernziele
 - Organisatorisches
 - Literatur
 - Bezeichnungen und Konventionen
 - Einordnung und historische Entwicklung
 - Voraussetzungen
- Schaltwerke

- Einleitung und Grundbegriffe, Definitionen
- Speicherelemente
- Analyse
- Synthese
- Zusammenschaltung
- Transformationen
- Zustandskodierung
- Zustandsminimierung
- Realisierung, Beispiele
- Zeitverhalten
 - Zeitverhalten von Schaltnetzen
 - Modellierung der Gatter- und Leitungsverzögerungen
 - Statische Timing-Analyse (STA)
 - Zeitverhalten von Schaltwerken
 - Metastabilität

Literatur

- Hoffmann, Dirk: Grundlagen der technischen Informatik, Carl Hanser Verlag 2007
- Schiffmann, Wolfram; Schmitz, Robert: Technische Informatik, in 3 Bänden. 3. Auflage Springer Verlag, 1996
- Veendrick, Harry: Nanometer CMOS ICs, Springer 2008
- Rabaey, Jan; Chandrakasan, Anantha; Nikolic, Borivoje: Digital Integrated Circuits, A Design Perspective, 2nd edition, Prentice Hall 2003
- Beuth, Klaus: Elektronik 4. Digitaltechnik, 13. Auflage, Vogel Verlag und Druck 2003

I.1.9.2 Rechnerstrukturen

Lehrveranstaltung	Rechnerstrukturen
Dozent(en)	Dennis Säring
Hörtermin	2
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.5
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	E-Learning

Lernziele

Die Studierenden ...

- kennen grundlegende Rechnerarchitekturkonzepte, die beschreiben, wie verschiedene Baugruppen von Rechnern zusammenarbeiten und wie sich unterschiedliche Rechner-systeme voneinander unterscheiden
- haben die Funktionselemente von Rechnern mit ihren typischen Systemeigenschaften und deren Abbildung auf ein bestimmtes Architekturmodell gelernt
- können das Zusammenwirken der beteiligten Hardware- und Softwarekonzepte im Rahmen einer Aufgabe zur Informationsverarbeitung einschätzen
- besitzen ein Verständnis für Ansätze zur Steigerung der Systemleistung insbesondere unter Berücksichtigung der Aspekte von Parallelität

Inhalt

- Entwicklung der Rechnerarchitekturen
- Grundlegende Rechnerstrukturen
- Architekturkonzepte
- Mikroprogrammierung
- Moderne parallele und nicht-sequentielle Architekturkonzepte
- Mehrprozessorsysteme
- Aktuelle und zukünftige Entwicklungen

Literatur

- Martin: Einführung in die Rechnerarchitektur, Fachbuchverlag Leipzig, 2003
- Oberschelp, Gossen: Rechneraufbau und Rechnerstrukturen, Verlag Oldenbourg 1998
- van de Goor: Computer Architecture and Design, Verlag Addison Wesley, 1989
- Müller-Schloer, Schmitter: RISC-Workstation Architekturen, Verlag Springer 1991

- Ungerer: Datenfluß-Rechner, Verlag Teubner, 1993

I.1.10 UNIX und Shell-Programmierung

B044 UNIX und Shell-Programmierung

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B044
Bezeichnung	UNIX und Shell-Programmierung
Lehrveranstaltung(en)	B044a UNIX & Shell Programmierung B044a Übg. UNIX & Shell-Programmierung
Verantwortliche(r)	M.Sc. Helga Karafiat
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul setzt auf den konzeptionellen Inhalten des Moduls "Programmstrukturen 1" und auf der dort erworbenen Programmiererfahrung auf. Es kann mit anderen fortgeschrittenen Modulen zur Software-Technik kombiniert werden, insbesondere mit "Systemnaher Programmierung" und den Themengebieten Betriebssysteme, Compilerbau, Skriptsprachen und Webentwicklung. Zudem schafft es Grundlagen und Verständnis für die Verarbeitung und Analyse großer Datenmengen (Data Science / Big Data).
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Vorausgesetzt werden Kenntnisse der Grundkonzepte imperativer Programmiersprachen hinsichtlich der Umsetzung algorithmischer Grundstrukturen und die Fähigkeit zur Erstellung von vollständigen Programmen begrenzter Komplexität.
Dauer	1
Lernziele	Nach Abschluss des Moduls kennen die Studierenden die grundlegende Funktionsweise von Unix-Systemen und sind mit der Verwendung von Skriptsprachen und Tools in diesem Umfeld vertraut.

Sie kennen die Vor- und Nachteile von Skriptsprachen bei der Software-Entwicklung und haben ein Gefühl für Automation von wiederkehrenden Aufgaben und für die Verarbeitung von großen Datenmengen entwickelt. Ferner kennen sie die regulären Ausdrücke zur Verarbeitung von Texten und deren Mächtigkeit und Grenzen. Sie können reguläre Ausdrücke sicher auf Problemstellungen anwenden. Sie haben zudem ein Verständnis für Datenströme entwickelt und die einfache und elegante Art der Kombinierbarkeit von Programmen, insbesondere an Hand von Filtern und Pipes, verinnerlicht.

Sie kennen sowohl die Flexibilität als auch die Fehleranfälligkeit von dynamischen Sprachen und haben eine Vorstellung davon, wann und wie sich die Produktivität beim Arbeiten mit Skriptsprachen im Vergleich zu kompilierten Sprachen verändert.

Die Studierenden finden sich durch den Umgang mit Unix-Systemen nun auch in einem Umfeld zurecht, in dem ihnen lediglich eine textbasierte Konsole zur Interaktion mit einem System zur Verfügung steht.

I.1.10.1 UNIX & Shell Programmierung

Lehrveranstaltung	UNIX & Shell Programmierung
Dozent(en)	Helga Karafiat
Hörtermin	2
Häufigkeit	jährlich
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assigm.
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, Tafel

Lernziele

Die Studierenden ...

- schätzen die Vorteile und Gefahren von Skriptsprachen in der Software-Entwicklung am Beispiel der Unix-Shells sh (dash) und bash ab.
- wenden reguläre Ausdrücke praktisch für die Verarbeitung von Texten und Auszeichnungssprachen an und kennen deren Mächtigkeit und Grenzen.
- arbeiten mit Filtern und Pipes und kombinieren diese zu einfachen Programmen beziehungsweise Skripten.
- haben ein Grundverständnis über interne Abläufe im Unix-Kern bei der Prozessverwaltung.
- haben ein Verständnis für die sequentielle Verarbeitung, wie Filterung und Auswertung, von großen Datenmengen in Textform.

Inhalt

- Unix
 - Systemstruktur
 - Shell Kommandos
 - Dateisystem und Rechteverwaltung
 - Filter und Pipelines
 - Skriptprogrammierung mit der Shell
 - POSIX-Konformität und nützliche Erweiterungen durch die bash
- Reguläre Ausdrücke
 - Aufbau und Zusammensetzung
 - Tools zur Verwendung von regulären Ausdrücken unter Unix (grep, sed)
 - Mächtigkeit und Grenzen von regulären Ausdrücken

- Suchen, Zerlegen und Editieren von Texten mit regulären Ausdrücken
- POSIX-konforme reguläre Ausdrücke und GNU-Erweiterungen
- Skriptsprachen
 - Einfache Shell-Programme (dash, bash)
 - Systematisches Kombinieren kleiner Programme
- Das make-System
- Prozessverwaltung

Literatur

- Karafiat, Helga: Unix & Shell-Programmierung Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~kar/unix-vorlesung/>
- Kofler, Michael: Linux: Das umfassende Handbuch, Rheinwerk Computing, 2017, ISBN: 978-38-36258-54-8
- Dietze, Martin: Praxiskurs Unix-Shell, O'Reilly Verlag GmbH & Co. KG; 2011, ISBN: 978-38-97215-65-8
- Robbins, Arnold; Beebe, Nelson H.F.: Klassische Shell-Programmierung, O'Reilly Verlag GmbH & Co. KG; 2006, ISBN: 978-38-97214-41-5
- Kernighan, Brian W.; Pike, Rob: UNIX-Werkzeugkasten: Programmieren mit UNIX, Hanser Fachbuch, 1986, ISBN-13: 978-34-46142-73-2
- Friedl, Jeffrey E. F.: Reguläre Ausdrücke, O'Reilly Verlag GmbH & Co. KG, 2007, ISBN: 978-38-97217-20-1
- Dougherty, Dale; Robbins, Arnold: sed & awk, O'Reilly and Associates, 1997, ISBN: 978-15-65922-25-9

I.1.10.2 Übg. UNIX & Shell-Programmierung

Lehrveranstaltung	Übg. UNIX & Shell-Programmierung
Dozent(en)	Malte Heins
Hörtermin	2
Häufigkeit	jährlich
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assigm.
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, studentische Arbeit am Rechner, Tutorien

Lernziele

Die Studierenden ...

- wenden praktisch die Inhalte aus der Vorlesung an.
- beherrschen den praktischen Umgang mit der Shell und den gängigsten UNIX Kommandos.
- erlangen die Fähigkeit zur Erstellung kleiner nichttrivialer Shell-Programme durch das Arbeiten mit Pipes und Filtern.

Inhalt

Bearbeitung von Übungsaufgaben parallel zum Stoff der Vorlesung in Zweiergruppen mit Abnahme und Diskussion der Lösungen.

Literatur

- Unterlagen zur Übung im Web
- siehe auch Vorlesung UNIX und Shell-Programmierung

I.1.11 Übertragungstechnik

B032 Übertragungstechnik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B032
Bezeichnung	Übertragungstechnik
Lehrveranstaltung(en)	B032a Übertragungstechnik
Verantwortliche(r)	Prof. Dr.-Ing. Carsten Burmeister
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Das Modul ist mit dem Modul "Physik und Elektrotechnik" und weiteren Modulen aus dem Bereich der technischen Informatik zu kombinieren.
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Es werden Kenntnisse aus dem Modul Physik und Elektrotechnik vorausgesetzt.
Dauer	1

Lernziele

Nach Abschluss des Moduls verfügen die Studierenden über ein Verständnis für Signalformen, -verzerrungen und -verarbeitung bei der Übertragung analoger und diskreter Signale. Hierfür verfügen sie über Kenntnisse hinsichtlich der Anwendung elektrischer Schaltungen in der Nachrichtenübertragung. Ferner verfügen sie über das Wissen zu Maßnahmen zur Qualitätssicherung bei der Signalübertragung. So verstehen sie Übertragungsvorgänge in physikalischen und logischen Netzstrukturen.

I.1.11.1 Übertragungstechnik

Lehrveranstaltung	Übertragungstechnik
Dozent(en)	Carsten Burmeister
Hörtermin	2
Häufigkeit	jährlich
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assigm.
Semesterwochenstunden	6
ECTS	5.0
Prüfungsform	Klausur + ggf. Bonus
Sprache	deutsch/englisch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, studentische Arbeit am Rechner, Tafel

Lernziele

Die Studierenden ...

- besitzen ein Verständnis für Signalformen, -verzerrungen und -verarbeitung bei der Übertragung analoger und diskreter Signale.
- können elektrische Schaltungen in der Nachrichtenübertragung anwenden.
- besitzen die Kenntnis hinsichtlich der Maßnahmen zur Qualitätssicherung bei der Signalübertragung.
- kennen physikalische und logische Übertragungsnetzstrukturen.

Inhalt

- Signalformen: Signalanalyse und -synthese
 - Sinusförmige Signale
 - Nicht-sinusförmige periodische Signale
 - Nicht-periodische Signale
- Filterstrukturen
 - Tiefpass und Hochpass 1. Ordnung
 - Tiefpass und Hochpass 2. und höherer Ordnung
 - Bandpass und Bandsperre 2. und höherer Ordnung
- Schwingkreise
 - Reihenschwingkreis
 - Parallelschwingkreis
 - Frequenzgänge, Ortskurven und Bodediagramme
- Modulationstechnik
 - Amplitudenmodulation

- Frequenz- und Phasenmodulation
- Kombination von Modulationsarten
- Nachrichtenmodelle
 - Nachrichtenkanal
 - Kommunikationsnetz
 - ISO-OSI-Kommunikationsmodell
 - Funktions- und Zeitabläufe Kommunikationsnetzen
- Datensicherung in gestörten Übertragungskanälen
 - Entropie, Transinformation, Irrelevanz und Äquivokation
 - Abtastung und Quantisierung
 - Bit-sequentielle Übertragung
 - Paritätssicherung und zyklische Redundanzcodes
- Übertragungstechniken
 - Simplex-, Halbduplex-, Vollduplexübertragung
 - Basisbandübertragung
 - Hochfrequente Übertragung: ASK, FSK, PSK und Kombinationen
 - Multiplexechniken: TDM, FDM, OFDM

Literatur

- Hagmann, G.: Grundlagen der Elektrotechnik. Aula-Verlag, 2000 (7. Auflage)
- Führer, A.; Heidemann, K.; Nerretter, W.: Grundgebiete der Elektrotechnik, Bd. 2: Zeitabhängige Vorgänge. Hanser-Verlag, 1990
- Papula, L.: Mathematik für Ingenieure, Bd. 2. Vieweg, 2000 (9. Auflage)
- Tanenbaum, A. S.: Computernetzwerke. Prentice-Hall International, 2003 (4. Auflage)
- Meyer, M.: Kommunikationstechnik. Vieweg-Teubner, 2008 (3. Auflage)
- Paul, S.: Grundlagen der Elektrotechnik und Elektronik 2: Elektromagnetische Felder und ihre Anwendungen. Springer-Verlag, 2019 (2. Auflage)

I.1.12 Deskriptive Statistik und Grundlagen der Linearen Algebra

B019 Deskriptive Statistik und Grundlagen der Linearen Algebra

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B019
Bezeichnung	Deskriptive Statistik und Grundlagen der Linearen Algebra
Lehrveranstaltung(en)	B019a Grundlagen der Linearen Algebra B019a Deskriptive Statistik
Verantwortliche(r)	Prof. Dr. Franziska Bönte
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Das Modul "Deskriptive Statistik & Grundlagen der Linearen Algebra" ist ein Einführungsmodul. Zusammen mit dem Modul "Analysis", stellt es die Grundlage für nahezu alle quantitativ ausgerichteten weiter führenden Module und Veranstaltungen des Studienverlaufs dar.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Vorausgesetzt wird ein solides Schulwissen der Mathematik und mindestens durchschnittliche mathematische Begabung.
Dauer	1

Lernziele

In diesem Modul werden grundlegende mathematische Kenntnisse aus den Bereichen lineare Algebra und deskriptiven Statistik, wie sie als Grundlage für ein quantitativ ausgerichtetes

Studium unerlässlich sind, vermittelt. Der Fokus liegt auf der Vektor- und Matrizenrechnung, linearen Gleichungssystemen und statistischer Datenanalyse.

Die Lernenden sind in der Lage ausgewählte Problemstellungen aus den Bereichen Technik, Naturwissenschaft, Ökonomie und Informatik, mittels der im Modul vermittelten mathematischen und statistischen Methoden zu modellieren und analysieren. Die Lernenden können für die vermittelten Inhalte praxisrelevante Anwendungsbeispiele benennen. Die Lernenden können eigenständig Lösungsmethoden für ausgewählte Problemstellungen auswählen, die Lösungsmethodik bis zum Ergebnis durchführen und die erhaltenen Ergebnisse kritisch bewerten.

I.1.12.1 Grundlagen der Linearen Algebra

Lehrveranstaltung	Grundlagen der Linearen Algebra
Dozent(en)	Andreas Haase
Hörtermin	2
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.5
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Handout, Overheadfolien, Tafel

Lernziele

Nach dem erfolgreichen Besuch der Vorlesung sind die Lernenden in der Lage ...

- lineare algebraische Gleichungssysteme mittels des Gauß-Algorithmus in die Lösbarkeitskategorien (eindeutig lösbar, unendlich viele Lösungen, unlösbar) einzuteilen und ggfs. die Lösung anzugeben.
- die Techniken und Methoden der Vektorrechnung anzuwenden.
- die Techniken und Methoden der Matrixrechnung anzuwenden.
- die Determinante einer niedrigdimensionalen Matrix zu berechnen und den Zusammenhang der Determinante zur Lösungstheorie linearer Gleichungssysteme herzustellen
- einfache technische oder ökonomische Systeme mittels der Techniken und Methoden der linearen Algebra zu modellieren und aus der ermittelten Lösung der mathematischen Formulierung das System quantitativ zu beurteilen.

Inhalt

- Lineare algebraische Gleichungssysteme
 - Gauß-Algorithmus
 - Systematisierung des Lösungsverhaltens
 - Unterbestimmte Systeme
- Matrixrechnung
 - Matrixalgebra
 - Inverse Matrix
 - Matrixgleichungen
 - Zusammenhang mit linearen Gleichungssystemen
- Determinanten
 - Definition

- Zusammenhang mit linearen Gleichungssystemen
- Vektorrechnung
 - Geometrische Vektoren
 - Rechenregeln
 - Lineare (Un-)Abhängigkeit
 - Rang einer Matrix
 - Nochmal Gleichungssysteme, Rangkriterium

Literatur

- PAPULA, Lothar:
Mathematik für Ingenieure und Naturwissenschaftler,
Band 2, Teil I. 13. Aufl. Wiesbaden: Vieweg + Teubner Verlag 2012
- HELM, Werner; PFEIFER, Andreas; OHSER, Joachim:
Mathematik für Wirtschaftswissenschaftler.
1. Aufl. München: Carl Hanser Verlag 2011
- GRAMLICH, Günter:
Lineare Algebra: Eine Einführung.
1. Aufl. München: Carl Hanser Verlag 2011
- TESCHL, Gerald; TESCHL, Susanne:
Mathematik für Informatiker,
Band 1: Diskrete Mathematik und lineare Algebra.
3. Aufl. Heidelberg: Springer Verlag 2008
- FISCHER, Gerd:
Lineare Algebra: Eine Einführung für Studienanfänger.
18. aktualisierte Aufl. Wiesbaden: Springer Verlag 2014

I.1.12.2 Deskriptive Statistik

Lehrveranstaltung	Deskriptive Statistik
Dozent(en)	Franziska Bönte
Hörtermin	2
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.5
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, E-Learning, Gastreferenten, interaktive Entwicklung und Diskussion von Modellen, Online-Aufbereitung, Tafel, Tutorien

Lernziele

Nach der Lehrveranstaltung können die Studierenden ...

- Statistische Daten verdichten und graphisch aussagekräftig darstellen
- Wesentliche Aussagen über Daten anhand geeigneter Kennzahlen treffen und interpretieren
- Die Ableitung von Regressionsformeln verstehen und komplexe Regressions- und deskriptive Zeitreihenanalysen abgestimmt auf den jeweiligen Datensatz durchführen und interpretieren
- sicher im Umgang mit Meß- und Indexzahlen agieren

Inhalt

Im Rahmen der beschreibenden / deskriptiven Statistik werden folgende Themen behandelt:

- Begrifflichkeiten
- Lage- und Streuungsmaße
- Abhängigkeitsmessung bei qualitativen, komperativen und quantitativen Merkmalen insbesondere Regressionsanalyse
- Deskriptive Zeitreihenanalyse mit Trend-, Saison- und Restkomponentenschätzung nach unterschiedlichen Methoden
- Meß- und Indexzahlen

Literatur

- Christensen, B.; Christensen, S.; Missong, M.: Statistik klipp & klar; 2019; Springer Gabler Verlag
- Bamberg, G.; Baur, F; Krapp, M: Statistik; 18. Auflage; 2017; De Gruyter Oldenbourg Verlag; München

- Missong, Martin; Aufgabensammlung zur deskriptiven Statistik; 2005; 7. Auflage; Verlag R. Oldenbourg, München.
- Schneider, Wolfgang; Kornrumpf, J.; Mohr, Walter; Statistische Methodenlehre --
- Definitions- und Formelsammlung zur deskriptiven und induktiven Statistik mit Erläuterungen; 1993; Verlag Oldenbourg, München.

I.1.13 Halbleiterphysik und Optik

B254 Halbleiterphysik und Optik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B254
Bezeichnung	Halbleiterphysik und Optik
Lehrveranstaltung(en)	B254a Optik B254a Atom- und Halbleiterphysik
Verantwortliche(r)	Dr. Hendrik Glowatzki
Zuordnung zum Curriculum	Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul „Halbleiterphysik & Optik“ ist ein Einführungsmodul. Zusammen mit dem Modul „Mechanik und Elektrotechnik“, stellt es die Grundlage für zahlreiche weiterführende technisch orientierte Module, wie zum Beispiel „Konstruktionstechnik“, „Fertigungstechnik“ oder „Verfahrenstechnik“ dar.
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Das Modul setzt elementare Kenntnisse der Schulphysik, insbesondere aus der Mechanik, wie sie auch im Modul „Mechanik und Elektrotechnik“ vermittelt werden, voraus. Ebenso müssen elementare Kenntnisse der Mathematik, insbesondere Differential- und Integralrechnung einer Funktion einer Variablen, Algebra und Vektorrechnung vorliegen.
Dauer	1

Lernziele

In diesem Modul werden die grundlegenden physikalischen Kenntnisse aus den Bereichen Atomphysik, Halbleiterphysik und Optik, die für das Verständnis von technischen Prozessen, z.B. aus den Bereichen Fertigungs- und Verfahrenstechnik, erforderlich sind, vermittelt. Die Lernenden können in technischen Systemen die physikalischen Prinzipien benennen und diese Systeme nach den physikalischen Prinzipien einteilen. Sie können physikalische und physikalisch-mathematische Methoden auf ausgewählte technische Problemstellungen anwenden und charakteristische Eigenschaften technischer Systeme aus den physikalischen Gesetzen ableiten.

I.1.13.1 Optik

Lehrveranstaltung	Optik
Dozent(en)	Andreas Haase
Hörtermin	2
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	1.5
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	

Lernziele

Die Lernenden können nach dem erfolgreichen Besuch der Vorlesung ...

- die vorgestellten physikalischen Begriffe und Gesetze der Optik selbständig erklären und zueinander in Beziehung setzen, bzw. gegeneinander abgrenzen.
- für ausgesuchte Aufgaben aus der Optik selbständig eine Lösungsstrategie entwickeln, in dem sie die dargestellte Problematik in den richtigen Kontext aus der Vorlesung einordnen.
- Aufgaben unter Anwendung der erlernten physikalischen und mathematischen Mittel und Methoden, eigenständig lösen.
- das Ergebnis einer gelösten Aufgabe kritisch bewerten und daraus Schlüsse und Folgerungen ziehen.

Inhalt

Reflexionsgesetz

- Reflexionsgesetz
- Ebene und sphärische Spiegel

Brechungsgesetz

- Brechungsgesetz von Snellius
- Totalreflexion und Lichtleiter
- Dispersion
- Dünne Linsen
- Abbildungsfehler

Interferenz

- Interferenz am Doppelspalt
- Vielstrahlinterferenz

- Lichtwellen
- Dünne Schichten
- Kohärenz

Beugung

- Das Huygenssche Prinzip
- Beugung am Einzelspalt
- Beugung am Gitter
- Beugung an der Lochblende

Polarisation

- Polarisation
- Polarisationsfilter
- Polarisationsverfahren

Optische Geräte

- Das Auge
- Die Lupe
- Das Mikroskop
- Das Fernrohr
- Der Laser

Literatur

Halliday, Resnick, Walker: Physik, Bachelor Edition, Wiley-VCH (2013)

Kersten (Hrsg.), Tipler: Physik für Studierende der Naturwissenschaften und Technik, Springer Spektrum (2019)

Kuypers: Physik für Ingenieure und Naturwissenschaftler, Band 2: Elektrizität, Optik, Wellen, Wiley-VCH (2012)

Meschede: Gerthsen Physik, Springer Spektrum (2015)

I.1.13.2 Atom- und Halbleiterphysik

Lehrveranstaltung	Atom- und Halbleiterphysik
Dozent(en)	Hendrik Glowatzki
Hörtermin	2
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	3.5
Prüfungsform	Klausur + ggf. Bonus
Sprache	deutsch
Lehr- und Medienform(en)	

Lernziele

Die Lernenden können nach dem erfolgreichen Besuch der Vorlesung ...

- die vorgestellten physikalischen Begriffe und Gesetze aus der Atom- und Halbleiterphysik selbständig erklären und zueinander in Beziehung setzen, bzw. gegeneinander abgrenzen.
- für ausgesuchte Aufgaben aus der Atom- und Halbleiterphysik selbständig eine Lösungsstrategie entwickeln, in dem sie die dargestellte Problematik in den richtigen Kontext aus der Vorlesung einordnen.
- Aufgaben unter Anwendung der erlernten physikalischen und mathematischen Mittel und Methoden, eigenständig lösen.
- das Ergebnis einer gelösten Aufgabe kritisch bewerten und daraus Schlüsse und Folgerungen ziehen.

Inhalt

- Atomphysik
 - Das Atommodell von Thomson
 - Der Streuversuch von Rutherford
 - Das Atommodell von Rutherford
 - Der Franck-Hertz-Versuch
 - Das Bohrsche Atommodell
 - Schalenstruktur der Atomhülle
- Quantenmechanik
 - Elektromagnetische Wellen
 - Der Photoeffekt
 - Schrödingergleichung und Wellenfunktion
 - Der Tunneleffekt

- Die Heisenbergsche Unschärferelation
- Welle-Teilchen-Dualismus
- Halbleiterphysik
 - Festkörper
 - Bändermodell
 - Löcherleitung
 - Driftgeschwindigkeit und Beweglichkeit
 - Ladungsträgerdichte und Fermienergie
 - Der pn-Übergang
 - Diode
 - Transistor
- Magnetismus und Materie

Literatur

- Hering, Martin, Stohrer - Physik für Ingenieure, Springer (2012)
- Tipler, Mosca: Physik für Studierende der Naturwissenschaften und Technik, Springer Spektrum (2019)
- Meschede: Gerthsen Physik, Springer Spektrum (2015)
- Ibach, Lüth - Festkörperphysik (2009)

I.1.14 Algorithmen und Datenstrukturen

B040 Algorithmen und Datenstrukturen

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B040
Bezeichnung	Algorithmen und Datenstrukturen
Lehrveranstaltung(en)	B040a Algorithmen und Datenstrukturen B040b Übg. Algorithmen & Datenstrukturen
Verantwortliche(r)	M.Sc. Christian Uhlig
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul setzt unmittelbar auf den Inhalten des Moduls „Programmstrukturen 2“ auf und eignet sich damit als Weiterqualifikation im Anschluss an „Programmstrukturen 2“ und das „Programmierpraktikum“. Es kann ergänzend mit fortgeschrittenen Modulen zur Software-Technik kombiniert werden, insbesondere mit „Software-Design“, „Fortgeschrittene Objektorientierte Programmierung“ und „Systemnahe Programmierung“.
Semesterwochenstunden	5
ECTS	5.0
Voraussetzungen	Vorausgesetzt werden theoretische und praktische Grundkenntnisse in der Programmiersprache Java, die insbesondere auch die Abbildung abstrakter Datentypen per Interfaces und abstrakter Klassen und die Verwendung generischer Typen umfassen.
Dauer	1

Lernziele

Nach Abschluss des Moduls sind sich die Studierenden der Einflussfaktoren auf den Speicherbedarf einer Datenstruktur und auf die Laufzeit der darauf ausgeführten Algorithmen bewusst. Sie können Laufzeit und Speicherbedarf systematisch auf Ebene des Programmcodes und aufgrund empirischer Untersuchung analysieren und vergleichend beurteilen, wobei sie zwischen dem konstanten Faktor und dem Laufzeitwachstum in Abhängigkeit von der Problemgröße unterscheiden. Hierbei verstehen und nutzen sie Komplexitätsklassen insbesondere per Groß-O-Notation und können eigenen Programmcodes in typische Komplexitätsklassen einordnen.

Weiterhin sind sich die Studierenden der Bedeutung des Sortierens und Suchens in der Softwaretechnik bewusst und kennen die wesentlichen theoretischen Grundlagen dieser Probleme und entsprechender Algorithmen.

Sie können die Trennung in abstrakte Datentypen und zugehörige Implementierungen erläutern und gewinnbringend für eigene Softwareprojekte anwenden. Hierbei wählen sie abhängig vom Anwendungsfall zwischen typischen abstrakten Datentypen wie z.B. Listen und Verzeichnissen aus.

Ihnen sind die Konzepte, die Funktionsweise wichtiger Operationen und die damit verbundenen Eigenschaften typischer Implementierungen wie z.B. verketteter Listen, Arrays und Binärer Suchbäume bekannt, so dass sie ausgehend von den einzusetzenden Operationen geeignete Implementierungen für den verwendeten abstrakten Datentypen auswählen.

Ausgehend von den erworbenen Grundlagenkenntnissen nutzen die Studierenden den Java Collections Framework und wählen dabei zielgerichtet abstrakte Datentypen und Implementierungen des Frameworks aus.

Die Studierenden können die Motivation und die Grundlagen nebenläufiger Algorithmen am Beispiel von dynamischem Multithreading erläutern und die Konzepte auf die Programmiersprache Java übertragen und im Rahmen einfacher Problemstellungen einsetzen.

I.1.14.1 Algorithmen und Datenstrukturen

Lehrveranstaltung	Algorithmen und Datenstrukturen
Dozent(en)	Christian Uhlig
Hörtermin	3
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	3
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, interaktive Entwicklung und Diskussion von Modellen, Softwaredemonstration

Lernziele

Die Studierenden ...

- analysieren, diskutieren und vergleichen einfache Algorithmen und Datenstrukturen hinsichtlich ihres Bedarfs an Laufzeit und Speicher.
- differenzieren bei der Analyse von Algorithmen hinsichtlich best case, worst case und average case.
- differenzieren die Laufzeit von Algorithmen nach ihrem konstanten Faktor und ihrem Wachstum in Abhängigkeit von der Problemgröße.
- beurteilen die Laufzeit von Algorithmen ausgehend von Komplexitätsklassen in den Groß-O-, Groß-Omega- und Groß-Theta-Notationen.
- nennen und erläutern wesentliche Aspekte, Funktionsweisen und Eigenschaften von Algorithmen zum Suchen und Sortieren.
- erläutern die Differenzierung in abstrakte Datentypen und ihre Implementierung.
- nennen und erläutern typische abstrakte Datentypen wie Listen, Mengen, Verzeichnisse und Warteschlangen mit ihren Operationen und Anwendungsbereichen.
- nennen und erläutern Motivation, Funktionsweise und Eigenschaften typischer Implementierungen abstrakter Datentypen mit verketteten Listen, Arrays, Baumstrukturen und Hash-Tabellen.
- wählen zu einer gegebenen Problemstellung einen geeigneten abstrakten Datentypen nebst einer geeigneten Implementierung aus.
- wenden die abstrakten Datentypen und Implementierungen des Java Collections Frameworks an.
- erläutern die Grundkonzepte nebenläufiger Programmierung.
- wenden die Primitiven dynamischen Multithreadings bei der Formulierung von Programmen an.

Inhalt

- Analyse von Algorithmen
 - Laufzeit und Speicherbedarf
 - Groß-O / Groß-Omega / Groß-Theta Notationen
 - Amortisierte Laufzeitanalyse
 - Iterative vs rekursive Implementierungen
- Sortieren und Suchen
- Listenstrukturen
 - Verkettete Listen
 - Arraybasierte Listen
 - Skiplisten
- Baumstrukturen
 - Binäre Suchbäume
 - Balancierte Suchbäume: 2-3-Bäume
 - Balancierte Binäre Suchbäume: Rot/Schwarz-Bäume
 - Balancierte Binäre Suchbäume: AVL-Bäume
 - Spreizbäume
 - Tries
 - Arraybasierte Binäre Heaps
- Hash-Tabellen
- Abstrakte Datentypen und ihre Implementierung
 - Listen
 - Mengen
 - Verzeichnisse
 - Warteschlangen
- Java Collections Framework
- Nebenläufige Algorithmen

Literatur

- Uhlig, Christian: Algorithmen und Datenstrukturen, Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~uhl/aud.html>
- Sedgewick, Robert; Wayne, Kevin: Algorithms, 4th Edition, Addison-Wesley, 2011
- Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford: Introduction to Algorithms, 3rd Edition, The MIT Press, 2009

- Knuth, Donald E.: The Art of Computer Programming Vol. 1 Fundamental Algorithms, 3rd Edition, Addison-Wesley, 1997
- Knuth, Donald E.: The Art of Computer Programming Vol. 3 Sorting and Searching, 2nd Edition, Addison-Wesley, 1998
- Wirth, Niklaus: Algorithmen und Datenstrukturen, 5. Auflage, Teubner, 2013
- Aho, Alfred V.; Hopcroft, John E.; Ullman, Jeffrey D.: The Design and Analysis of Computer Algorithms, 1st Edition, Pearson, 1975
- Aho, Alfred V.; Hopcroft, John E.; Ullman, Jeffrey D.: Data Structures and Algorithms, Addison-Wesley, 1983
- Aho, Alfred V.; Ullman, Jeffrey D.: Foundations of computer science, Computer Science Press, 1992

I.1.14.2 Übg. Algorithmen & Datenstrukturen

Lehrveranstaltung	Übg. Algorithmen & Datenstrukturen
Dozent(en)	Malte Heins
Hörtermin	3
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, studentische Arbeit am Rechner, Tutorien

Lernziele

Die Studierenden ...

- wenden praktisch die Inhalte aus der Vorlesung an.
- vertiefen die Beherrschung der Programmiersprache Java und der objektorientierten Programmierung.
- erlangen die Fähigkeit zur Erstellung algorithmenorientierter Programme in Java.

Inhalt

Bearbeitung von Übungsaufgaben parallel zum Stoff der Vorlesung in Zweiergruppen mit Abnahme und Diskussion der Lösungen. Zusätzlich werden praxisrelevante Aspekte der Anwendungsentwicklung mit der Programmiersprache Java behandelt, die nicht Bestandteil der Vorlesung sind wie z. B. Dateieingabe und -ausgabe.

Literatur

- Unterlagen zur Übung im Web

I.1.15 Systemnahe Programmierung

B043 Systemnahe Programmierung

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B043
Bezeichnung	Systemnahe Programmierung
Lehrveranstaltung(en)	B043a Systemnahe Programmierung B043b Übg. Systemnahe Programmierung
Verantwortliche(r)	M.Sc. Christian Uhlig
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) IT-Ingenieurwesen (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul setzt auf den konzeptionellen Inhalten des Moduls „Programmstrukturen 1“ und der im Modul „Programmstrukturen 2“ erworbenen fortgeschrittenen Programmiererfahrung auf. Es kann mit anderen fortgeschrittenen Modulen zur Software-Technik kombiniert werden, insbesondere mit „Algorithmen und Datenstrukturen“, und schafft die notwendigen Voraussetzungen für Anschlussmodule (z.B. im Bereich der Computergrafik), die Kenntnisse in der Programmiersprache C erfordern.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Vorausgesetzt werden Grundkenntnisse in statisch getypten imperativen Programmiersprachen, die insbesondere charakteristische Datentypen und Kontrollstrukturen (Sequenz, Selektion, Iteration) umfassen. Diese Kenntnisse sollten mit gefestigter Programmierpraxis in einer entsprechenden Sprache verbunden sein.
Dauer	1
Lernziele	Nach Abschluss des Moduls kennen die Studierenden die Charakteristika und die wesentlichen Elemente der Programmiersprache C und können die Einsatzfelder der Sprache benennen. Sie wenden die erworbenen Kenntnisse zur Implementierung von Software in C praktisch an. Dabei

setzen sie typische Sprachkonzepte wie Zeiger nebst Zeigerarithmetik, Arrays, dynamische Speicherverwaltung, bitweise Operationen und Funktionszeiger sowie den Präprozessor und die Standard-Bibliothek in sicherer und bezogen auf den Anwendungsfall möglichst portabler Form ein.

Weiterhin sind die Studierenden sensibilisiert für die inhärenten Probleme systemnaher Programmierung insbesondere in Hinblick auf unsichere Sprachkonzepte, schwache Typprüfungen, Portabilitätsprobleme durch plattformabhängiges Verhalten und Konsequenzen von Programmfehlern. Sie setzen diese Sensibilisierung in besondere Sorgfalt und Präzision bei der Programmierung in C um.

Sie können die Funktionsweise eines Rechners und die damit verbundenen Kosten bei der Ausführung von Programmen einer Hochsprache insbesondere bezüglich der Umsetzung von Datentypen, der Ausführung von Schleifen und Verzweigungen und der Implementierung von Unterprogrammaufrufen grob darstellen und im Zuge eigener Entwicklungstätigkeit einfließen lassen.

I.1.15.1 Systemnahe Programmierung

Lehrveranstaltung	Systemnahe Programmierung
Dozent(en)	Christian Uhlig
Hörtermin	3
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, interaktive Entwicklung und Diskussion von Modellen, Softwaredemonstration

Lernziele

Die Studierenden ...

- formulieren Programme in der Programmiersprache C unter Berücksichtigung der Besonderheiten der Programmiersprache insbesondere in Hinblick auf undefiniertes Verhalten, Plattformabhängigkeiten und Unsicherheiten bestimmter Sprachkonstrukte (z.B. Zeigerarithmetik und fehlende Boundary Checks).
- erläutern in groben Zügen die Repräsentation von Daten und die Abläufe in einem Rechner bei der Ausführung von Anweisungen und Auswertung von Ausdrücken in einer höheren Programmiersprache, insbesondere im Rahmen von Unterprogrammaufrufen.

Inhalt

- Typische Elemente und Eigenschaften eines C-Programms
- Datentypen
 - Ganzzahl- und Aufzählungstypen
 - Fließkommatypen, Grundlagen von Fließkommazahlen
 - Strukturierte Typen
 - Vereinigungstypen
 - Zeigertypen
 - Arraytypen
- Funktionszeiger und ihre Anwendungsbereiche
- Konvertierungen
- Arrays und ihre Beziehung zu Zeigern
- Ausdrücke
 - Konstanten
 - Grundlegende Ausdrücke (Zuweisungen, Funktionsaufrufe, etc.)

- Arithmetische Ausdrücke
- Boolesche Ausdrücke, Vergleichsoperatoren, logische Operatoren
- Bitweise Operatoren
- Arbeit mit Zeigern und Zeigerarithmetik
- Vorrang und Assoziativität
- Anweisungen, insbesondere Verzweigungen und Schleifen
- Dynamische Speicherverwaltung
- Übersetzungsprozess und C-Präprozessor
- Funktionsaufrufe in Maschinen, Aufrufstapel
- Gefahren der Sprache C am Beispiel eines Buffer Overflows mit Manipulation der Rücksprungadresse

Literatur

- Uhlig, Christian: Systemnahe Programmierung, Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~uhl/sp.html>
- Harbison, Samuel; Steele, Guy L.: C - A Reference Manual, 5th edition, Prentice Hall, New Jersey, 2002
- Kernighan, Brian W.; Ritchie, Dennis M.: C Programming Language, Prentice Hall, New Jersey, 1998

I.1.15.2 Übg. Systemnahe Programmierung

Lehrveranstaltung	Übg. Systemnahe Programmierung
Dozent(en)	Malte Heins
Hörtermin	3
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, studentische Arbeit am Rechner, Tutorien

Lernziele

Die Studierenden ...

- wenden praktisch die Inhalte aus der Vorlesung an.
- beherrschen die Programmiersprache C und die maschinennahen Konzepte der Sprache.
- erlangen die Fähigkeit zur Erstellung maschinennaher Programme.
- erlangen das Verständnis über die Abläufe in einer Maschine bei der Ausführung von Sprachkonstrukten aus höheren Programmiersprachen, wie zum Beispiel die Laufzeitor-
ganisation bei Funktionsaufrufen.

Inhalt

Bearbeitung von Übungsaufgaben parallel zum Stoff der Vorlesung in Zweiergruppen mit Abnahme und Diskussion der Lösungen. Zusätzlich werden praxisrelevante Aspekte der Anwendungsentwicklung mit der Programmiersprache C behandelt, die nicht Bestandteil der Vorlesung sind.

Literatur

- Unterlagen zur Übung im Web
- siehe auch Vorlesung Systemnahe Programmierung

I.1.16 Rechnernetze

B037 Rechnernetze

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B037
Bezeichnung	Rechnernetze
Lehrveranstaltung(en)	B037a Rechnernetze B037b Prakt. Rechnernetze
Verantwortliche(r)	Dipl.-Ing. (FH) Ilja Kaleck
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul ist sinnvoll mit den Inhalten der Grundlagemodule "Informationstechnik" und "Programmstrukturen 1 und 2" zu kombinieren.
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Grundlegende Kenntnisse im Bereich der allgemeinen Informationstechnik; grundlegende Programmierkenntnisse in C, Objekt-Pascal (Delphi) oder Java erleichtern das Verständnis für Interprozesskommunikation im Rahmen gezeigter Beispielprogramme; Kenntnisse im Umgang mit aktuellen Desktop-Betriebssystemen (Windows, optional MacOS-X bzw. Linux) sind zur eigenständigen Durchführung praktischer Übungsanteile hilfreich.
Dauer	1

Lernziele

Nach Beendigung dieses Moduls verfügen die Studierenden über fundiertes Wissen über den Aufbau, den Betrieb und die Arbeitsweise moderner Rechnernetze (Computer Networks); dieses sowohl in technischer Hinsicht als auch in Bezug auf den Ablauf der Kommunikation zwischen Prozessen in Unternehmensnetzen bzw. dem Internet. Die Studierenden beherrschen allgemeine

Grundlagen der Datenkommunikation und kennen den Aufbau eines universellen Kommunikationsmodells, erlernt am Beispiel des OSI-Referenzmodells.

Vertiefendes Wissen haben sie bezüglich des Aufbaus und die Kommunikation in der Internet-Architektur (IPv4, IPv6). Hierbei verfügen sie über Kenntnisse hinsichtlich der Eigenschaften der verfügbaren Transportprotokolle und haben das grundlegende Verständnis zur Realisierung einfacher Interprozesskommunikation.

Sie kennen die für den Betrieb eines IP-basierten Netzes essentiell notwendigen Anwendungsprotokolle und können dieses Wissen auch als Basis für die Gestaltung eigener Anwendungen sinnvoll nutzen. Ferner verfügen sie über Kenntnisse hinsichtlich der Theorie und den praktischen Einsatz von Verzeichnisdiensten zur Verwaltung größerer Netze.

Darüber hinaus haben sie ein hinreichendes Verständnis für den technischen Aufbau und den Betrieb moderner Unternehmensnetze. Hierzu gehören fundierte Kenntnisse über die Eigenschaften aktueller Netztechnologien im Bereich Lokaler Netze (LANs) als auch drahtloser Netze (WLANs).

Sie kennen auch die Arbeitsweise der dabei eingesetzten Koppellemente und deren Vermittlungsstrategien zum Aufbau größerer Netzstrukturen bzw. des Internets.

Durch den praktischen Anteil des Moduls verfügen die Studierenden über ein vertieftes und punktuell auch signifikant ergänztes Wissen zuvor behandelter Lehrinhalte, eigenständig erlernt am eigenen PC-System (Server) im zugehörigen Schulungslabor. Sie verfügen auch über ein praxisnahes Verständnis über den realen Datenfluss in Netzen und können so typische Fehlersituationen im Rahmen der Kommunikation zwischen Anwendungen analysieren und eigenständig beheben. Diese Fähigkeit bildet eine wesentliche Grundlage für eine effiziente Entwicklung verteilter Anwendungen im Rahmen komplexer Softwareprojekte.

I.1.16.1 Rechnernetze

Lehrveranstaltung	Rechnernetze
Dozent(en)	Ilja Kaleck
Hörtermin	3
Häufigkeit	jedes Semester
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	E-Learning, Online-Aufbereitung, Softwaredemonstration

Lernziele

Die Studierenden erlangen ...

- ein grundlegendes Verständnis für den Aufbau einer herstellerneutralen Kommunikationsarchitektur (OSI).
- Kenntnisse über den Aufbau und die Funktion des Internet-Architekturmodells.
 - Kenntnis über IPv4-Adress- und Netzstrukturen.
 - Verständnis über die Arbeitsweise essentieller Anwendungsprotokolle.
 - Fähigkeit zum Verständnis des Ablaufs einfacher Interprozesskommunikation, u.a. als Basis für die Realisierung komplexerer verteilter Anwendungen.
 - die Arbeitsweise spezifischer Maßnahmen gegen den IPv4-Adressmangel im IPv4 (NAT, Proxyserver-Dienste) kennen.
 - Wissen über die Eigenschaften des neuen Internet-Protokolls Version 6 (IPv6) und Änderungen an bestehenden Internet-Protokollen (u. a. DNS, ICMP).
- Verständnis über den technischen Aufbau und den Betrieb Lokaler Netze (LANs).
 - Verständnis hinsichtlich des generellen Ablaufs der IP-Kommunikation in LANs.
 - Wissen um die Eigenschaften aktueller Netztechnologien (Schwerpunkt: Ethernet-Technik).
 - Kenntnisse zum Aufbau und Betrieb drahtloser Netze (IEEE 802.11 WLANs).
- Wissen um den technischen Aufbau von Netzstrukturen bzw. des Internets.
 - Wissen um die Aufgabe Funktionsweise der klassischen von Koppellemente in Netzen.
 - elementares Wissen um die Arbeitsweise praxisrelevanter Routingverfahren für kleinere und größere Netze (u. a. einfaches IP-Routing; hierarchisches Routing).
- Grundkenntnisse über den Aufbau und die Funktionsweise von Verzeichnisdiensten.

Inhalt

- Allgemeine Grundlagen und Begriffe
 - Allgemeine Strukturen in der Datenkommunikation
 - Protokolle und Protokollabläufe
 - Netztopologien und Klassifizierung von Übertragungsnetzen
- Das ISO-OSI Referenzmodell
 - Prinzip der Schichtenbildung und Schichtenfunktionen im Überblick
 - Datenfluss im Modell
 - Aktuelle Koppellemente zum Netzaufbau im Kontext der OSI-Modells
- Die Internet-Architektur
 - Historie, Architekturübersicht, Standardisierungen
 - IPv4-Adressstrukturen und Netzaufbau, Subnetting
 - UDP-/TCP-Kommunikation, Sockets bzw. Socket-Kommunikation
 - Betrachtung ausgewählter Anwendungsprotokolle (DNS, TELNET / SSH, SMTP, HTTP, ...)
 - Network Address Translation (NAT) und der Einsatz von Proxy-Servern
 - Einführung in das neue Internet Protocol Version 6 (IPv6)
 - * Adress- und Netzstruktur, Migrationshinweise
 - * Änderungen an höheren Protokollen in Bezug auf das IPv6
- Technik Lokaler Netze (LANs)
 - Ablauf der Kommunikation in IEEE 802 LANs (Layer-2, IP, inkl. DHCP)
 - Schwerpunktbetrachtung: Ethernet-Technik, Zugriffsverfahren und
 - Technische Umsetzungen (10Mbps / 100FE / 1GbE / 10GbE)
 - Überblick über andere LAN-Technologien
- Koppellemente und Vermittlungstechniken
 - Repeater, Brücken- bzw. Layer-2 Switching-Technologie
 - Virtuelle LANs (VLANs), Class-of-Services im LAN
 - Router bzw. IP-Routing, Link-State und Distanzvektor-Verfahren,
 - Hierarchisches Routing und IP-Multicasting
 - Drahtlose Netze nach IEEE 802.11,
 - * Struktur, Aufbau, Übertragungskonzepte, Sicherheitsbetrachtungen
- Verzeichnisdiente
 - Einführung und grundlegendes Konzept des X.500

- Herstellerspezifische Lösungen (Active Directory)
- Lightweight Directory Access Protocol (LDAP)

Literatur

- TANNENBAUM, Andrew S.:
Computer Netzwerke.
5. Aufl. München: Pearson Education, 2012, ISBN 978-3-86894-137-1
- KUROSE, James F.; ROSS, Keith W.:
Computer Netzwerke. Der Top-Down Ansatz.
6. Aufl. : Pearson Education, 2014, ISBN 978-3-86894-237-8
- HALSALL, Fred:
Computer Networking and the Internet.
5. Aufl. München: Addison-Wesley, 2005, ISBN 978-0321263582
- RECH, Jörg:
Ethernet. Technologien und Protokolle für die Computervernetzung.
2. Aufl. Heidelberg: dPunkt-Verlag, 2007, ISBN 978-3-936931-40-2
- RECH, Jörg:
Wireless LANs. 802.11-WLAN-Technologie und praktische Umsetzung im Detail.
4. Aufl. Heidelberg: dPunkt-Verlag, 2012, ISBN 978-3-936931-75-4
- BADACH, Anatol; HOFFMANN, Erwin:
Technik der IP-Netze. Funktionsweise, Protokolle und Dienste.
2. Aufl. München: Hanser, 2007, ISBN 978-3446215016
- DAVIES, Joseph:
Understanding IPv6. Covers Windows 8 and Windows Server 2012.
3rd Edition: Microsoft Press, 2012, ISBN 978-0-7356-5914-8
- SCHÄFER, Günther:
Netzwerksicherheit. Algorithmische Grundlagen und Protokolle.
Heidelberg: dPunkt-Verlag, 2003, ISBN 3-89864-212-7
- SPERZEL Christian:
Netzwerksicherheit. Schützen Sie Ihr Netzwerk vor dem Zugriff anderer
Online-Videotrainig, Video2brain GmbH, 2014,
- BUEROSSE, Jörg:
Sichere E-Mails. Verschlüsselung und digitale Signatur unter Windows, Linux, OS X,
iOS und Android.
Online-Videotrainig, Video2brain GmbH, 2014
- FRISCH; HÖLZEL; LINTERMANN; SCHAÄFER:
Vernetzte IT-Systeme.
6. Aufl.:Bildungsverlag EINS, 2013, ISBN 978-3-8237-1141-4
- GRABA, Jan:
An Introduction to Network Programming with Java, Java 7 Compatible
3rd Edition: Springer-Verlag, 2013, ISBN 978-1-4471-5253-8

- CIUBOTARU, Bogdan ; MUNTEAN, Gabriel-Miro:
Advanced Network Programming - Principles and Techniques. Network Application Programming with Java.
Springer-Verlag, 2013, ISBN 978-1-4471-5291-0
- HAROLD, Elliotte Rusty:
Java Network Programming. Developing Networked Applications.
4th Edition, OReilly Media, 2013, ISBN 978-1-44935-767-2
- KLÜNTER, Dieter; LASER, Jochen:
LDAP verstehen, OpenLDAP einsetzen. Grundlagen und Praxiseinsatz.
2. Aufl. Heidelberg: dPunkt-Verlag, 2007, ISBN 978-3-89864-263-7

I.1.16.2 Prakt. Rechnernetze

Lehrveranstaltung	Prakt. Rechnernetze
Dozent(en)	Ilja Kaleck
Hörtermin	3
Häufigkeit	jedes Semester
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	E-Learning, Online-Aufbereitung, Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden erlangen ...

- die Fähigkeit zum praktischen Umgang mit der Internet-Technologie am eigenen PC.
 - die Fähigkeit zum Anschluss von Systemen an ein Unternehmensnetz.
 - die Fähigkeit zur grundlegenden Konfiguration des Internet-Protokolls (IPv4, IPv6).
 - das Verständnis für Sicherheitsrichtlinien auf Multi-User Systemen (Windows, Linux).
 - die Fähigkeit zur Analyse und Behebung typischer Fehlersituationen im Rahmen der Kommunikation von Anwendungen und Systemen im Netz.
 - die Fähigkeit zur Konfiguration grundlegender Internet-Dienste (u. a. DNS, HTTP, FTP).
- das Verständnis für Lösungsansätze aktueller Techniken zur Unix-/Windows Integration in heterogenen Unternehmensnetzen (NFS, SAMBA, X-Windows).
- das Verständnis über aktuelle Konzepte zur Benutzer- und Rechteverwaltung in Netzen.
 - die Fähigkeit zur Benutzerverwaltung mittels eines Domänenkonzeptes (Windows).
 - die Fähigkeit zur Einrichtung von Verzeichnisdiensten (LDAP, Active Directory).
- die Grundkenntnisse zum praktischen Einsatz von Virtualisierungstechniken auf dem Desktop.
 - die Fähigkeit zur Einrichtung einfacher IP-Routingfunktionen auf einem System.
- das Verständnis über den praktischen Aufbau und Betrieb eines WLANs und dessen interne Kommunikationsabläufe (inkl. Sicherheitsbetrachtungen).
- die Fähigkeit zum Einsatz eines LAN-Analyzers zur Analyse von Kommunikationsabläufen zwischen Anwendungen sowie zur Fehleranalyse in LANs und WLANs.
- grundlegende Kenntnisse digitaler Sprachübertragung in Netzen mittels der Voice-over-IP (VoIP) Technik (Wahlthema).

- grundlegende Kenntnisse zu Streaming-Media Technik und den Real-Time Protokollen zur Übertragung multimedialer Inhalte in Netzen (Wahlthema).

Inhalt

Durchführung eines Laborpraktikums durchgängig individuell am eigenen PC-System unter Einsatz dedizierter Wechselfestplatten (Teilnehmer; Arbeitsgruppe)

- Einrichtung eines Server-Betriebssystems und Konfiguration der grundlegenden Kommunikationsprotokolle (IPv4, IPv6).
 - Nutzung typischer Internetdienstprogramme und Betrachtung der dabei verwendeten Protokolle.
- Einsatz von Techniken zur Unix/Windows-Integration (NFS, SAMBA, X-Windows, Unix mit Posix-ACLs)
- Nutzung einfacher Benutzer- und Rechteverwaltung im Netz (Domänenkonzept).
- Einsatz von Virtualisierungstechniken auf dem Desktop
 - Aufbau einer lokalen Netzinfrastruktur und Einrichtung des lokalen IP-Routings (inkl. NAT)
 - Grundlegende Firewall-Konfiguration
- Einrichten und Arbeiten mit aktuellen Verzeichnisdiensten
 - Aufbau einer eigenen Verzeichnisstruktur (Directory)
 - Formulierung von Suchanfragen an Verzeichnisdienste (Active Directory, LDAP-Server)
- Konfiguration grundlegender Internet-Serverdienste (DNS, FTP, HTTP, Proxy-Server, TELNET / SSH)
 - Nutzung der SSH Port-Forwarding Funktion
- Protokollanalyse und Fehlersuche im LAN mit einem LAN-Analyzer
 - Nutzung einer Remote-Probes zur verteilten LAN-Analyse im Netz.
 - Einfache LAN-Performance Messungen
- Konfiguration einer Arbeitsstation in einem Wireless-LAN (Adhoc und Infrastrukturnetz)
 - Analyse des drahtlosen Daten- und Kontrollverkehrs mit einem WLAN-Analyzer
- Einrichtung eines Voice-over-IP (VoIP) Clients (Wahlaufgabe)
 - Betrachtung dabei genutzter VoIP-Technologien und Übertragungsprotokolle
 - Einsatz eines LAN-Analyzers zur VoIP-Übertragungsanalyse
- Einführung in die Multi-Media Übertragung in Netzen (Wahlaufgabe)
 - Einrichtung eines eines aktuellen Streaming-Servers
 - Betrachtung der beteiligten Realtime-Übertragungsprotokolle
- Weitere Wahlthemen nach Aktualität.

Literatur

- RECH, Jörg:
Wireless LANs. 802.11-WLAN-Technologie und praktische Umsetzung im Detail.
4. Aufl. Heidelberg: dPunkt-Verlag, 2012, ISBN 978-3-936931-75-4
- BADACH, Anatol:
Voice-over-IP. Grundlagen, Protokolle, Anwendungen, Migration, Sicherheit.
4. Aufl. München: Hanser, 2009, ISBN 978-3-446-41772-4
- LIU/MATTHEW/PARZIALE/DAVIS/FORRESTER/BRITT:
TCP/IP Tutorial and Technical Overview (PDF). 8th. Ed. 2006: IBM-Redbook Serie.
<http://www.redbooks.ibm.com/redbooks/> Aktualisierungsdatum 29.06.2014
- GROUPER IEEE 802.11: Aktuelle Spezifikationen zu IEEE 802.11.
<http://standards.ieee.org/getieee802/802.11.html> Aktualisierungsdatum 29.06.2014
- IETF: Internet-Draft Dokumente und aktuelle RFCs.
<http://www.ietf.org/> - Aktualisierungsdatum 29.06.2014
- CISCO SYSTEMS: Internetworking Technology Handbook.
<http://www.cisco.com/c/en/us/td/docs/internetworking/technology/handbook/itodoc.html>
Aktualisierungsdatum 29.06.2014
- SPERZEL, Christian:
Netzwerksicherheit. Schützen Sie Ihr Netzwerk vor dem Zugriff anderer
Online-Videotrainig, Video2brain GmbH, 2014,
<https://www.video2brain.com/de/videotraining/netzwerksicherheit> - Aktualisierungsdatum 29.06.2014
- BUEROSSE, Jörg:
Sichere E-Mails. Verschlüsselung und digitale Signatur unter Windows, Linux, OS X, iOS und Android.
Online-Videotrainig, Video2brain GmbH, 2014,
<https://www.video2brain.com/de/videotraining/sichere-e-mails> - Aktualisierungsdatum 29.06.2014
- DIVERSE:
Schulungskurse zum Thema "Virtualisierung".
Online-Videotrainig, Video2brain GmbH, 2013,
<https://www.video2brain.com/de/search.htm?searchentry=Virtualisierung> - Aktualisierungsdatum 29.06.2014
- WOWZA MEDIA SYSTEMS:
Online Dokumentation zur "Wowza Streaming Engine"
<http://www.wowza.com/forums/content.php?188-documentation> - Aktualisierungsdatum 29.06.2014

I.1.17 Elektronik

B048 Elektronik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B048
Bezeichnung	Elektronik
Lehrveranstaltung(en)	B048a Elektronik
Verantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul "Elektronik" baut auf den in den Modulen "Analysis", "Deskriptive Statistik & Grundlagen der Linearen Algebra", "Physik und Elektrotechnik" und "Übertragungstechnik" erworbenen Kenntnissen und Fähigkeiten auf. Die im Modul "Elektronik" erworbenen Kompetenzen stellen die Grundlagen für zum Beispiel die Module "Halbleiterschaltungstechnik" und "Systemtheorie" dar. Das Modul kann sinnvoll mit Modulen zu elektrotechnischen Grundlagen und digitaler Elektronik bzw. Halbleiterschaltungstechnik kombiniert werden. Das Modul hat einen direkten Bezug zum Studiengangsziel des Studiengangs "Bachelor Technische Informatik" hardwarenahe Kompetenzen in technischen Anwendungen der Informatik zu erwerben.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Es werden Kompetenzen aus den Modulen "Analysis", "Deskriptive Statistik & Grundlagen der Linearen Algebra" "Physik und Elektrotechnik" sowie "Einführung in Digitaltechnik" vorausgesetzt.
Dauer	1

Lernziele

Nach dem Kennerlernen der grundlegenden Eigenschaften und Berechnungsverfahren linearer elektrischer Netzwerke (Lehrveranstaltung "Grundlagen der Elektrotechnik") sollen die Studierenden im Modul "Elektronik" systematisch an die allgemeinen Entwurfs- und Analyseverfahren elektronischer Schaltungen herangeführt werden. Das Modul vermittelt Kenntnisse

und Fertigkeiten, die für sicheren Umgang mit modernen Entwurfs- und Analysewerkzeugen notwendig sind, wobei theoretische Grundlagen durch interaktive Beispiele und Bezugnahme auf Datenblätter real existierender Bauelemente und Schaltungen sowie durch praxisbezogene Übungsaufgaben ergänzt werden. Die Studierenden lernen, aus den zurückliegenden Lehrveranstaltungen bereits bekannte Bauelemente in Form von Modellen und Ersatzschaltbildern darzustellen und zu analysieren, sowie die Toleranzen und Temperaturabhängigkeiten mit in die Entwurfs- und Analysebetrachtungen einzubeziehen. Mit einer Einführung in die Methoden und Verfahren der Systemanalyse lernen die Studierenden schließlich die systematische Sicht auf elektronische Schaltungen und Systeme kennen und sind in der Lage, mit unterschiedlichen Typen zeitkontinuierlicher Signale (Gleichstrom, harmonische Erregung, Ausgleichsvorgänge, sowie deren Überlagerungen) umzugehen. Gleichzeitig wird eine Basis für das Verständnis der grundlegenden Funktionsweise moderner Simulations- und Analysewerkzeuge (P Spice, Mat-Lab) gelegt, auf der weitere Module aufbauen.

I.1.17.1 Elektronik

Lehrveranstaltung	Elektronik
Dozent(en)	Sergei Sawitzki
Hörtermin	3
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	5.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, Tafel

Lernziele

Die Studierenden ...

- bekommen eine Einsicht in die theoretischen Grundlagen der Elektronik
- lernen Methoden und Werkzeuge für Entwurfs und Analyse von elektronischen Systemen kennen
- lernen, das reale Verhalten von elektronischen Bauteilen (Toleranzen, Temperaturabhängigkeiten usw.) zu berücksichtigen
- erwerben die Fähigkeit, einfache analoge und digitale Systeme aus dem Bereich der technischen Informatik zu begreifen, zu spezifizieren und zu entwerfen

Inhalt

- Passive Bauelemente und Schalter
 - Übersicht der elektronischen Bauelemente, E-Normreihen
 - Widerstand, Kondensator, Spule
 - Schalter
- Thermisches Verhalten von Bauelementen
 - Einleitung und Grundbegriffe, Wärmestromkreis
 - Aspekte des thermischen Verhaltens
 - Erwärmung und Abkühlung, Einsatz von Kühlkörpern
- Lineare Netzwerke bei Gleichstrom
 - Strom- und Spannungsquellen, Grundstromkreis
 - Bestimmung des Arbeitspunktes
 - Allgemeine Netzwerkanalyse
- Lineare Netzwerke bei zeitabhängiger Erregung

- Wechselstrom und Wechselspannung
- Passive Filter, Schwingkreise und Resonatoren
- Wechselstrombrücken
- Ausgleichsvorgänge
- Anwendungen der Systemanalyse
 - Einleitung
 - Fourier-Reihen, Klirrfaktor
 - Fourier-und Laplace-Transformation
 - Blockschaltbild-Algebra
 - Vierpoltheorie

Literatur

- Tietze, Ulrich; Schenk, Christoph: Halbleiterschaltungstechnik, 16. Auflage Springer Verlag, 2016
- Stiny, Leonhard: Handbuch passiver elektronischer Bauelemente, Franzis Verlag 2007
- Hering, Ekbert; Bessler, Klaus; Gutekunst, Jürgen: Elektronik für Ingenieure und Naturwissenschaftler, Springer Verlag 2005
- Lunze, Klaus: Berechnung elektrischer Stromkreise, 15. Auflage Huss Medi 1990
- Horowitz, Paul; Hill, Winfield: Die Hohe Schule der Elektronik. Teil 1: Analogtechnik, 8. Auflage Elektor-Verlag 2006
- Schiffmann, Wolram; Schmitz, Robert: Technische Informatik, in 3 Bänden. 3. Auflage Springer Verlag, 1996
- Brauer, Harry; Lehmann, Constans; Lindner, Helmut: Taschenbuch der Elektrotechnik und Elektronik, 9. Auflage Hanser Fachbuchverlag, 2008

I.1.18 Ingenieurmathematik

B046 Ingenieurmathematik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B046
Bezeichnung	Ingenieurmathematik
Lehrveranstaltung(en)	B046a Ingenieurmathematik
Verantwortliche(r)	Dr. Hendrik Glowatzki
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Das Modul "Ingenieurmathematik" baut auf den in der Veranstaltung "Analysis" und "Deskriptive Statistik & Grundlagen der Linearen Algebra" erworbenen Kenntnissen und Fähigkeiten auf. Die im Modul "Ingenieurmathematik" erworbenen Kompetenzen stellen die Grundlage für zum Beispiel die weiterführenden Module "Regelungstechnik", "Einführung in die Robotik", "Elektrotechnik" oder "Diskrete Systeme" dar.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Es werden Kenntnisse in Analysis und Linearer Algebra in einem Umfang vorausgesetzt, wie sie etwa durch den erfolgreichen Besuch der entsprechenden Veranstaltungen aus den Modulen zur Grundlagen der Mathematik erworben werden können (B001 und B019). Für den Teil numerische Mathematik sind erste Programmiererfahrungen hilfreich.
Dauer	1

Lernziele

In diesem Modul werden weiterführende mathematische Kenntnisse aus den Bereichen der mehrdimensionalen Analysis und der numerischen Mathematik, wie sie für ein quantitativ ausgerichtetes Studium unerlässlich sind, vermittelt.

Die Lernenden sind in der Lage ausgewählte Problemstellungen der Ingenieurmathematik, mittels der im Modul vermittelten mathematischen Methoden zu modellieren und analysieren. Die

Lernenden können für die vermittelten Inhalte praxisrelevante Anwendungsbeispiele benennen. Die Lernenden können eigenständig Lösungsmethoden für ausgewählte Problemstellungen auswählen, die Lösungsmethodik bis zum Ergebnis durchführen und die erhaltenen Ergebnisse kritisch bewerten. Die Lernenden sind in der Lage zu entscheiden in welchen Fällen eine exakte analytische Methodik zum Erfolg führt und in welchen Fällen eine numerische Methode angewendet werden muss.

I.1.18.1 Ingenieurmathematik

Lehrveranstaltung	Ingenieurmathematik
Dozent(en)	Hendrik Glowatzki
Hörtermin	3
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	5.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, studentische Arbeit am Rechner, Tafel

Lernziele

Die Veranstaltung gliedert sich in zwei sukzessive Teile.

Teil 1: Höhere Analysis.

Die Lernenden können nach dem erfolgreichen Besuch ...

- eine skalare Funktion von mehreren Variablen einmal und mehrfach nach allen Variablen ableiten.
- das totale Differenzial einer mehrdimensionalen skalaren Funktion bilden und seine Bedeutung erklären.
- die mehrdimensionale Kettenregel und die implizite Differenziation anwenden.
- die Lage der lokalen Extrema einer mehrdimensionalen skalaren Funktion, mit und ohne Nebenbedingung, berechnen.
- Flächen und Volumenintegrale berechnen.
- ausgewählte Klassen gewöhnlicher Differenzialgleichungen erster und zweiter Ordnung nach Lösungsmethode klassifizieren und mittels der vorgestellten Verfahren lösen.

Teil 2: Numerische Mathematik

Die Lernenden können nach dem erfolgreichen Besuch ...

- die Notwendigkeit für numerische Verfahren anführen.
- die prinzipiellen Beschränkungen und Fehler numerischer Verfahren aufzählen und darlegen.
- Nullstellen von skalaren nichtlinearen Funktionen mittels der vorgestellten Methoden näherungsweise bestimmen und die Güte der Approximation mittels Fehleranalyse untersuchen.
- lineare Gleichungssysteme numerisch mittels direkter und iterativer Verfahren lösen und die Güte des erhaltenen Ergebnisses mittels Fehleranalyse evaluieren.
- eine gegebene Menge von Datenpunkten interpolieren. Insbesondere können die Lernenden das einfache Interpolationspolynom berechnen und sind in der Lage eine lineare

stückweise Interpolierende und einen stückweise definierten kubischen Spline zu berechnen.

- eine gegebene Menge von Datenpunkten mittels einer Menge von Ansatzfunktionen approximieren. Dabei können sie das zu Grunde liegende Minimierungsproblem selbständig formulieren und lösen.
- eine gegebene eindimensionale Funktion numerisch differenzieren und integrieren und die Fehler der Algorithmen bewerten und die Fehler des Ergebnisses berechnen.
- eine gegebene gewöhnliche Differenzialgleichung erster Ordnung mittels verschiedener Einschrittverfahren näherungsweise lösen und den Fehler des Ergebnisses unter Verwendung der Fehleranalyse abschätzen.
- Programmiererfahrene Lernende können die dargestellten Algorithmen in entsprechende Computercodes übersetzen.

Inhalt

Teil 1: Höhere Analysis

- Funktionen mehrerer Variablen
- Differenzialrechnung für Funktionen mehrerer Variablen
 - partielle Differenziation
 - Kettenregel und Richtungsableitung
 - Extremwerte mit und ohne Nebenbedingung
- Integralrechnung
 - Doppelintegral
 - Dreifachintegral
- Gewöhnliche Differenzialgleichungen 1. und 2. Ordnung

Teil 2: Numerische Mathematik

- Rechnerarithmetik; Gleitpunktzahlen und Fehlerrechnung
- Numerische Lösung von Nullstellenproblemen
 - Bisektionsverfahren
 - Fixpunktiteration
 - Newtonverfahren
- Numerische Lösung linearer Gleichungssysteme
 - Gauß-Algorithmus und Dreieckszerlegung
 - Fehlerrechnung
 - Iterative Verfahren
- Interpolation. Polynome und kubische Splines.

- Approximation. Lineare Ausgleichsrechnung.
- Numerisches differenzieren und integrieren
- Anfangswertprobleme gewöhnlicher Differenzialgleichungen

Literatur

- PAPULA, Lothar:
Mathematik für Ingenieure und Naturwissenschaftler. Band 2
- 13. durchgesehene Aufl. Wiesbaden: Vieweg + Teubner 2012
- PAPULA, Lothar:
Mathematik für Ingenieure und Naturwissenschaftler. Band 3
- 6. überarbeitete und erweiterte Aufl. Wiesbaden: Vieweg + Teubner 2011
- KNORRENSCHILD, Michael:
Numerische Mathematik: Eine beispielorientierte Einführung.
- 5. aktualisierte Aufl. München: Carl Hanser Verlag 2013
- SCHWARZ, Rudolf; KÖCKLER, Norbert:
Numerische Mathematik.
- 8. aktualisierte Aufl. Wiesbaden: Vieweg + Teubner 2011

I.1.19 Lineare Algebra

B045 Lineare Algebra

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B045
Bezeichnung	Lineare Algebra
Lehrveranstaltung(en)	B045a Lineare Algebra
Verantwortliche(r)	Dr. Andreas Haase
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) IT-Ingenieurwesen (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul "Lineare Algebra" baut auf den in der Veranstaltung "Grundlagen der Linearen Algebra" aus dem Modul "Deskriptive Statistik und Grundlagen der Linearen Algebra" erworbenen Kenntnissen und Fähigkeiten auf. Die im Modul "Lineare Algebra" erworbenen Kompetenzen stellen die Grundlage für zum Beispiel die weiterführenden Module "Grundlagen der Computergrafik", "Systemmodellierung" oder "Bildbearbeitung und -analyse" dar.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Das Modul setzt grundlegende Kenntnisse der linearen Algebra voraus, wie sie zum Beispiel im Modul "Deskriptive Statistik und Grundlagen der Linearen Algebra" vermittelt werden.
Dauer	1

Lernziele

In diesem Modul werden weiterführende mathematische Kenntnisse aus dem Bereich der linearen Algebra, wie sie für ein quantitativ ausgerichtetes Studium unerlässlich sind, vermittelt. Der Schwerpunkt liegt dabei auf der Vektorraumtheorie und der analytischen Geometrie.

Die Lernenden sind in der Lage ausgewählte Problemstellungen aus den Bereichen Naturwissenschaft, Technik und Informatik, mittels der im Modul vermittelten mathematischen Methoden

zu modellieren und analysieren. Die Lernenden können für die vermittelten Inhalte praxisrelevante Anwendungsbeispiele benennen. Die Lernenden können eigenständig Lösungsmethoden für ausgewählte Problemstellungen auswählen, die Lösungsmethodik bis zum Ergebnis durchführen und die erhaltenen Ergebnisse kritisch bewerten

I.1.19.1 Lineare Algebra

Lehrveranstaltung	Lineare Algebra
Dozent(en)	Andreas Haase
Hörtermin	3
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	5.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Tafel

Lernziele

Nach dem erfolgreichen Besuch der Veranstaltung können die Studierenden ...

- die Determinante einer Matrix beliebiger Dimension berechnen und den Zusammenhang zur Lösungstheorie linearer Gleichungssysteme herstellen.
- die Vektorraumaxiome nennen und eine gegebene Menge mit Verknüpfungen darauf überprüfen ob diese ein Vektorraum (über \mathbb{R} oder \mathbb{C}) ist.
- Die Definition eines Unterraums nennen; Teilmengen von Vektorräumen darauf überprüfen ob diese Unterräume sind.
- das Konzept der linearen Abhängigkeit von Vektoren erklären; Teilmengen von Vektorräumen auf lineare Abhängigkeit überprüfen.
- die Definition einer Basis nennen. Teilmengen von Vektorräumen darauf überprüfen ob diese eine Basis sind.
- die Definition eines Skalarproduktes nennen; verschiedene lineare Abbildungen auf Vektorräumen darauf überprüfen ob diese ein Skalarprodukt sind.
- die Definition einer Norm nennen; den Zusammenhang zwischen Skalarprodukt und Norm nennen.
- Die Definition einer Orthonormalbasis nennen; eine Orthonormalbasis aus einer gegebenen Basis konstruieren (Gram-Schmidt-Verfahren).
- die Parameter und Koordinatendarstellung von Geraden und Ebenen formulieren; Lagebeziehungen zwischen linearen geometrischen Objekten berechnen; Lagebeziehungen zwischen linearen und einfachen nichtlinearen Geometrischen Objekten berechnen.
- die Definition einer linearen Abbildung nennen; lineare Abbildungen mittels Matrix-Vektor-Schreibweise ausdrücken. Eigenschaften gegebener linearer Abbildungen bestimmen.
- die Definition einer affinen Abbildung nennen; affine Abbildungen mittels Matrix-Vektor-Schreibweise ausdrücken. Eigenschaften gegebener affiner Abbildungen bestimmen.
- Koordinatentransformationen als affine Abbildung durchführen; die affine Abbildung einer Koordinatentransformation berechnen; aktive und passive Koordinatentransforma-

tionen unterscheiden.

- das charakteristische Polynom einer Matrix aufstellen; die Eigenwerte einer Matrix berechnen; die Eigenvektoren einer Matrix berechnen.
- eine Matrix diagonalisieren.
- bestimmte Funktionen einer Matrix berechnen.

Inhalt

- Wiederholung: Grundlagen der linearen Algebra
- Determinanten
 - der Entwicklungssatz von Laplace
 - lineare Gleichungssysteme
- Vektorräume
 - Definition, Beispiele und Eigenschaften
 - Unterräume
 - Lineare Abhängigkeit, Basis und Dimension
- Euklidische und unitäre Vektorräume
 - Skalarprodukt und Norm
 - Orthogonalität
 - Orthogonal- und Orthonormalbasen
- Analytische Geometrie
 - Darstellung von Geraden und Ebenen
 - Lagebeziehung zwischen linearen geometrischen Objekten
 - Einfache nichtlineare Objekte am Beispiel
- Abbildungen
 - Lineare Abbildungen
 - Affine Abbildungen
 - Koordinatentransformationen
- Eigenwerte und Eigenvektoren
 - Charakteristisches Polynom, Eigenwerte, Eigenvektoren
 - Diagonalisierung
 - Matrixfunktionen

Literatur

- GRAMLICH, Günter M.:
Lineare Algebra: Eine Einführung.
3. aktualisierte Aufl. München: Carl Hanser Verlag 2011
- FISCHER, Gerd:
Lernbuch Lineare Algebra und Analytische Geometrie.
1. Aufl. Wiesbaden: Vieweg + Teubner Verlag 2011
- FARIN, Gerald; HANSFORD, Dianne:
Lineare Algebra: Ein geometrischer Zugang,
Springer Verlag 2003
- FISCHER, Gerd:
Lineare Algebra: Eine Einführung für Studienanfänger.
18., aktualisierte Aufl. Wiesbaden: Springer Verlag 2013
- LIESEN, Jörg; MEHRMANN, Volker:
Lineare Algebra: Ein Lehrbuch über die Theorie mit Blick auf die Praxis.
1. Aufl. Wiesbaden: Vieweg + Teubner Verlag 2011
- ZIESCHANG, Heiner:
Lineare Algebra und Geometrie.
1. Aufl. Stuttgart, Teubner Verlag 1997

I.1.20 Fortgeschrittene Objektorientierte Programmierung

B057 Fortgeschrittene Objektorientierte Programmierung

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B057
Bezeichnung	Fortgeschrittene Objektorientierte Programmierung
Lehrveranstaltung(en)	B057a Fortgeschrittene Objektorientierte Programmierung B057b Übg. Fortgeschrittene Objektorientierte Programmierung
Verantwortliche(r)	M.Sc. Christian Uhlig
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul setzt unmittelbar auf den Inhalten des Moduls „Programmstrukturen 2“ auf und eignet sich damit als Weiterqualifikation im Anschluss an „Programmstrukturen 2“ und das „Programmierpraktikum“. Es kann ergänzend insbesondere mit dem Modul „Software-Design“ kombiniert werden.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Vorausgesetzt werden gefestigte theoretische und praktische Kenntnisse in objektorientierter Programmierung im Allgemeinen und in der Programmiersprache Java im Speziellen. Dies sollte auch mindestens Einstiegskenntnisse zu generischen Typen (Java Generics) umfassen.
Dauer	1

Lernziele

Nach Abschluss des Moduls beherrschen die Studierenden den methodisch fundierten praktischen Umgang mit fortgeschrittenen objektorientierten Sprachkonzepten am Beispiel von Java. Sie sind in der Lage, vorgegebene generische Typen zu nutzen und eigene generische Typen zu deklarieren sowie in problemadäquater Weise Funktionswerte und Prinzipien funktionaler Programmierung in objekt-orientierten Sprachen anzuwenden. Die Teilnehmer erkennen Nutzen und Probleme nebenläufiger Programmierung und werden in die Lage versetzt, Threads und Konzepte zur Thread-Synchronisation anzuwenden. Ergänzend erlernen die Studierenden die Grundzüge der Programmierung mit Reflexion und können die Java Reflection API in ihren fundamentalen Einrichtungen nebst Java-Annotationstypen praktisch anwenden.

I.1.20.1 Fortgeschrittene Objektorientierte Programmierung

Lehrveranstaltung	Fortgeschrittene Objektorientierte Programmierung
Dozent(en)	Christian Uhlig
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, interaktive Entwicklung und Diskussion von Modellen, Softwaredemonstration

Lernziele

Die Studierenden ...

- nennen und erläutern die methodischen Grundlagen von objektorientierten Programmiersprachen am Beispiel von Java.
- wenden fortgeschrittene Konzepte objektorientierter Programmiersprachen wie parametrische Polymorphie, Mehrfachvererbung und Funktionswerte zur Konstruktion wiederverwendbarer Softwarekomponenten an.
- nennen und erläutern die Grundlagen generischer Datentypen und ihre Korrespondenz mit Konzepten objektorientierter Sprachen (Schnittstellen, abstrakte Klassen, konkrete Klassen, Polymorphie, Verträge).
- entwerfen und implementieren generische Datentypen.
- definieren Funktionswerte per anonymer Klassen, Lambda-Ausdrücke und Methodenreferenzen.
- wenden Prinzipien funktionaler Programmierung in objektorientierten Sprachen am Beispiel von Java Streams an.
- nennen und erläutern Motivation, Grundlagen und Herausforderungen nebenläufiger Programmierung.
- wenden Primitiven nebenläufiger Programmierung in Java an (Erzeugen von Threads, Thread-Kommunikation / Synchronisation, usw.).
- nennen und erläutern die Vorzüge funktionaler Programmierung für die parallele Auswertung von Berechnungen am Beispiel von Java Streams.
- definieren Stream-Pipelines unter Berücksichtigung der Anforderungen und Konsequenzen einer nebenläufigen Auswertungsstrategie.
- nennen und erläutern die Grundzüge von Reflection in Programmiersprachen am Beispiel von Java.
- wenden die Java Reflection API i.V.m. Annotationstypen an.

Inhalt

- Generische Typen / Java Generics
- Verschachtelte Typen
- Funktionswerte in OO-Sprachen (Funktionale Interfaces, Lambda-Ausdrücke, Methodenreferenzen)
- Funktionale Programmierung mit Java Streams
- Nebenläufigkeit (Threads, Racing Conditions, Synchronisation, Waitsets, volatile Variablen, Java Memory Model, nebenläufige Auswertung von Streams, Executors, Futures, Fork-Join-Tasks, Thread-sichere Collections)
- Reflection, Annotationstypen

Literatur

- Uhlig, Christian: Fortgeschrittene Objektorientierte Programmierung, Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~uhl/foop.html>
- Gosling, James; Joy, Bill; Steele, Guy; Bracha, Gilad; Buckley, Alex: The Java Language Specification, Java SE 8 Edition, 2015
- Meyer, Bertrand: Objektorientierte Softwareentwicklung, Hanser, München, 1990
- Bloch, Joshua: Effective Java: A Programming Language Guide, 3rd Edition, Addison-Wesley, 2017
- Urma, Raoul-Gabriel; Fusco, Mario; Mycroft, Alan: Java 8 in Action: Lambdas, Streams, and Functional-Style Programming, Manning Publications, 2014
- Ullenboom, Christian: Java ist auch eine Insel, 12. Auflage, Galileo Press GmbH, 2016
- Louden, Kenneth C.: Programming Languages: Principles and Practice, 2nd Edition, Thomson Learning, 2002

I.1.20.2 Übg. Fortgeschrittene Objektorientierte Programmierung

Lehrveranstaltung	Übg. Fortgeschrittene Objektorientierte Programmierung
Dozent(en)	Malte Heins
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, studentische Arbeit am Rechner, Tutorien

Lernziele

Die Studierenden ...

- wenden praktisch die Inhalte aus der Vorlesung an.

Inhalt

Bearbeitung von Übungsaufgaben parallel zum Stoff der Vorlesung in Zweiergruppen mit Abnahme und Diskussion der Lösungen. Zusätzlich werden praxisrelevante Aspekte der fortgeschrittenen OOP behandelt, die nicht Bestandteil der Vorlesung sind.

Literatur

- Unterlagen zur Übung im Web
- siehe auch Vorlesung Fortgeschrittene Objektorientierte Programmierung

I.1.21 Echtzeitsysteme

B101 Echtzeitsysteme

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B101
Bezeichnung	Echtzeitsysteme
Lehrveranstaltung(en)	B101a Echtzeitsysteme B101b Prakt. Echtzeitsysteme B101a Interface-Technologie
Verantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Informatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	<p>Das Modul "Echtzeitsysteme" baut auf den in den Modulen "Algorithmen und Datenstrukturen" und (im geringeren Maße) "Systemnahe Programmierung" erworbenen Kenntnissen und Fähigkeiten auf und vertieft diese im Bezug auf Echtzeitanwendungen. Das Modul kann sinnvoll durch die Module, die allgemeine Aspekte von Betriebssystemen (ohne besonderen Augenmerk auf Echtzeitfähigkeit) vermitteln, ergänzt werden. Auch eine Vertiefung durch die Module, die aktuelle industrielle Standards und Anwendungen betrachten, ist denkbar. Das Modul ist fachübergreifend in allen informatik-affinen Studiengängen einsetzbar, die Kompetenzen zum Echtzeit-Betrieb von Rechnersystemen vermitteln. Das Modul hat einen direkten Bezug zum Studiengangsziel des Studiengangs "Bachelor Technische Informatik" softwarenahe Kompetenzen in technischen Anwendungen der Informatik zu erwerben.</p>
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Es werden Kenntnisse der Programmiergrundlagen (Datentypen, Programmstrukturen), Basiskenntnisse aus der Informations- und Digitaltechnik (Zahlendarstellungen, Ablauf der Befehlsausführung, Daten- und Steuerfluss) sowie Beherrschung der grundlegenden Methoden und Verfahren der linearen Algebra (Matrixoperationen, Lösung von linearen Gleichungssystemen) vorausgesetzt.

Lernziele

Das Modul vermittelt Kenntnisse und Fertigkeiten, die für die Konzipierung, Entwurf, Inbetriebnahme sowie den Umgang mit Echtzeitsystemen benötigt werden. Dabei werden gleichermaßen Software- und Hardware-Konzepte betrachtet. Einerseits werden Methoden und Mechanismen vorgestellt, mit denen Systeme von nebenläufigen, kooperierenden oder konkurrierenden Prozessen modelliert und implementiert werden. Andererseits werden hardwaretechnische Voraussetzungen für Echtzeit-Betrieb diskutiert, sowie die Unterschiede, die ein Echtzeit-Betriebssystem im Vergleich mit gewöhnlichem Betriebssystem aufweist. Schließlich werden relevante Aspekte der Ereigniserfassung und -verarbeitung im einem Rechnersystem besprochen, untermauert durch Einsatzbeispiele aus der industriellen Praxis. Auch Sicherheits- und Zuverlässigkeitsaspekte werden betrachtet. Durch selbständiges Lösen einer praktischen Aufgabenstellung aus dem Bereich Echtzeitsysteme sollen die Studierenden ihren Lernerfolg überprüfen.

I.1.21.1 Echtzeitsysteme

Lehrveranstaltung	Echtzeitsysteme
Dozent(en)	Sergei Sawitzki
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	1.5
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Handout, Overheadfolien, Softwaredemonstration, Tafel

Lernziele

Die Studierenden ...

- bekommen die Kenntnisse vermittelt, die für die systemnahe Programmierung benötigt werden; insbesondere diejenigen Softwarekenntnisse, die für die Programmierung von Embedded Systems, Systemen mit kleinen Hardwareressourcen und Echtzeitsystemen erforderlich sind.
- erlernen Modellierungstechniken und -methoden bei Programmierung von nebenläufigen Prozessen, Modellierung und Anwendung von Prozesskommunikations- und Synchronisationsmechanismen; insbesondere die Fähigkeit, Aufgabenstellungen auf Systeme nebenläufiger (kooperierender und konkurrierender) Prozesse abzubilden und diese softwaretechnisch umzusetzen.
- bekommen eine Einführung in die Methodik der Programmierung paralleler Prozesse, unter besonderer Berücksichtigung von Echtzeitanforderungen vermittelt und erwerben die Befähigung, einfache Echtzeitsysteme zu konzipieren und zu realisieren. Neben Kenntnis wesentlicher theoretischer Konzepte wird besonders auch Fähigkeit zu deren praktischer Umsetzung, die im Projekt Echtzeitsysteme durchgeführt wird, vermittelt.
- weisen durch Entwurf und Realisierung eines Echtzeit-Multitasking-Programms die Kompetenz nach, derartige Systeme aus einem Anforderungskatalog zu entwerfen und praktisch umzusetzen.
- trainieren die Arbeitskoordination, Teamfähigkeit und soziale Kompetenz durch die im Projekt praktizierte Teamarbeit.

Inhalt

- Einleitung
 - Lernziele
 - Organisatorisches
 - Literatur
 - Bezeichnungen und Konventionen
 - Definition und Einordnung

- Historische Entwicklung
- Prozesse
 - Grundbegriffe
 - Technische Umsetzung
 - Aufgaben des Betriebssystems
 - Probleme und Lösungsansätze
- Kommunikationsmechanismen
 - Übersicht
 - Semaphore
 - Monitore
 - Mailbox-Kommunikation
 - Nachrichtenaustausch
 - Weitere Mechanismen
 - Äquivalenzen und Beispiele
- Modellierung
 - Einleitung
 - Flussdiagramme
 - Petri-Netze
 - Weitere Modellierungstechniken
- Scheduling
 - Einleitung
 - Strategien
 - Zeitverwaltung
 - Beispiele

Literatur

- Tanenbaum, Andrew: Moderne Betriebssysteme, Pearson Studium, 3. Auflage, 2009
- Witzak, Michael: Echtzeit Betriebssysteme, Franzis Verlag, 2000
- Baumgarten, Bernd: Petri-Netze, Wissenschaftsverlag, 1990
- Labrosse, Jean: MicroC/OS-II, CMP Books, 2002
- Quade, Jürgen; Mächtel, Michael: Moderne Realzeitsysteme kompakt, dpunkt.verlag, 2012

I.1.21.2 Prakt. Echtzeitsysteme

Lehrveranstaltung	Prakt. Echtzeitsysteme
Dozent(en)	Timm Bostelmann
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, studentische Arbeit am Rechner, Tafel

Lernziele

Nachdem Studierende die Veranstaltung besucht haben, können sie

- ausgehend von einer Aufgabenstellung in Form eines Anforderungskatalogs ein System nebenläufiger (kooperierender und konkurrierender) Prozesse entwerfen.
- ein solches System unter Verwendung eines Echtzeitbetriebssystems praktisch umsetzen.
- sich mit Hilfe eines Handbuches in die Schnittstelle eines Echtzeitbetriebssystems einarbeiten.
- ihre Lösung auf System- und Implementierungsebene in einer schriftlichen Ausarbeitung dokumentieren.

Inhalt

- Einführungsvorlesung
 - Motivation zur Veranstaltung
 - Beschreibung der Systemumgebung
 - Einführung in die Verwendung des Echtzeitkerns
- Einführungsaufgabe
 - Geführtes Erstellen eines Beispielprojektes
 - Teilweise geführte Programmierung eines Prozesses
 - Selbstständige Programmierung eines Prozesses
- Eine Aufgabe aus dem Umfeld Echtzeit, Multitasking, Simulation wird gestellt und steht den Studierenden als Anforderungskatalog zur Verfügung
 - Struktureller Programmentwurf
 - Kodierung und Test
 - Erstellung einer Dokumentation
 - Abnahme durch den Betreuer

Literatur

Labrosse, Jean: MicroC/OS-II, CMP Books, 2002

I.1.21.3 Interface-Technologie

Lehrveranstaltung	Interface-Technologie
Dozent(en)	Dennis Säring
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	1.5
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	E-Learning

Lernziele

Die Studierenden ...

- sind vertraut mit den grundlegenden Begriffen der Prozessdatenverarbeitung
- besitzen ein Verständnis für die sensorgestützte Aufnahme und Verarbeitung von Echtzeitdaten
- kennen die mathematischen Modelle für eine prädiktive Zustandsschätzungen auf Basis zurückliegender Messergebnisse
- können die Anforderungen an moderne BUS-Systeme im Kontext von komplexen Kommunikationsstrukturen (z.B. PKW, Flugzeug) einschätzen

Inhalt

- Grundlagen der Prozessdatenverarbeitung (PDV)
- Sensortechniken
- Strategien zur Fusionierung von Prozessdaten
- Prädiktionsmodell Kalman- und Partikel-Filter
- Moderne BUS-Systeme

Literatur

- Börcsök: Prozeßrechner- und Automation, Heise-Verlag, 1997
- Jacobsen: Einführung in die Prozeßdatenverarbeitung, Hanser-Verlag, 1996
- Wittgruber: Digitale Schnittstellen und BUS-Systeme, Vieweg-Verlag, 1999

I.1.22 Bildbearbeitung und -analyse

B097 Bildbearbeitung und -analyse

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B097
Bezeichnung	Bildbearbeitung und -analyse
Lehrveranstaltung(en)	B097b Prakt. Bildbearbeitung und -analyse B097a Bildbearbeitung und -analyse
Verantwortliche(r)	PD Dr. Dennis Säring
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul sollte in Verbindung mit dem Modul "Grundlagen der Computergrafik" gehört werden, da letzteres sich viel mit 2D-Algorithmen befasst und auch die Verfahren der Bildbearbeitung sich oftmals in 3D bewegen. Aufgrund aktueller Entwicklungen in der Forschung wird dieser Zusammenhang sich in naher Zukunft verstärken.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Grundlagen der Mathematik
Dauer	1

Lernziele

Lernziel ist es, dass erworbene Kompetenzen aus der Lehrveranstaltung *Bildbearbeitung und -analyse* schnell und effizient praktisch realisiert bzw. angewendet werden können. Dies wird durch das praktische Lösen kleinerer Aufgabenstellungen im betreffenden Praktikum erreicht.

I.1.22.1 Prakt. Bildbearbeitung und -analyse

Lehrveranstaltung	Prakt. Bildbearbeitung und -analyse
Dozent(en)	Hermann Höhne
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden...

- sammeln praktische Erfahrungen beim Implementieren von Algorithmen zur Bildbearbeitung
- vertiefen durch praktisches Umsetzen die in der zugehörigen Vorlesung theoretisch erläuterten Algorithmen zur Bildbearbeitung und -analyse
- üben das selbstständige Erweitern ihrer Kenntnisse in Programmiersprachen
- üben das selbstständige Einarbeiten in eine vorgegebene Bibliothek
- können diese Fähigkeiten im kommerziellen oder akademischen Umfeld praktisch nutzen

Inhalt

- Selbstständiges Einarbeiten in den C++17 Standard auf Basis des vorhandenen Hintergrundwissens um C
- Selbstständiges Einarbeiten in die Basisfunktionen von OpenCV bei minimaler Hilfestellung
- Implementierung von Algorithmen zur Bildbearbeitung und -analyse:
 - Punktweise Transformationen (Fensterung, Gamma-Korrektur)
 - Lineare Faltungsfiler und morphologische Filter
 - Affine Transformationen und Interpolation
 - Farb- und Texturanalyse
 - Grundlagen der Objekterkennung inklusive Segmentierung und Formerkennung

Literatur

I.1.22.2 Bildbearbeitung und -analyse

Lehrveranstaltung	Bildbearbeitung und -analyse
Dozent(en)	Dennis Säring
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	E-Learning

Lernziele

Die Studierenden ...

- haben ein Verständnis für die Struktur digitale Bilddaten
- erkennen die Möglichkeiten der Darstellung und Anpassung von digitalen Bildern
- können das Konzept der Bildkomposition und der Bildfilterung in praktischen Übungen umsetzen
- sind in der Lage Bilddaten in Frequenz- und Ortsbereich zu analysieren und zu bearbeiten
- verfügen über die theoretischen Grundlagen zur Registrierung von Bilddaten
- kennen unterschiedliche Ansätze Objekte in Bilddaten automatisch zu klassifizieren und zu segmentieren

Inhalt

- Einführung in die Bildbearbeitung
- Visualisierung und Bildanpassung
- Komposition und Filterung
- Fourier-Transformation und Frequenzfilter
- Lineare und nicht-lineare Registrierung
- Segmentierung und Texturanalyse
- Klassifikationsverfahren

Literatur

- Burger, Burge: Digitale Bildverarbeitung, Springer-Verlag 2005
- Handels: Medizinische Bildverarbeitung, Vieweg+Teubner 2009

I.1.23 Halbleiterschaltungstechnik

B068 Halbleiterschaltungstechnik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B068
Bezeichnung	Halbleiterschaltungstechnik
Lehrveranstaltung(en)	B068a Halbleiterschaltungstechnik B068b Übg. Elektronik und Halbleiterschaltungstechnik
Verantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul "Halbleiterschaltungstechnik" baut auf den in den Modulen "Elektronik" und "Rechnerstrukturen und Digitaltechnik" erworbenen Kenntnissen und Fähigkeiten auf. Die im Modul "Halbleiterschaltungstechnik" erworbenen Kompetenzen stellen die Grundlagen für zum Beispiel die Module "Großintegrierte Systeme" dar. Grundsätzlich ist eine Kombination mit den Modulen, die integrierte Schaltungen und Systeme auf höheren Abstraktionsebenen betrachten, sinnvoll. Das Modul hat einen direkten Bezug zum Studiengangsziel des Studiengangs "Bachelor Technische Informatik" hardwarenahe Kompetenzen in technischen Anwendungen der Informatik zu erwerben und praktisch einzusetzen.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Es werden Kompetenzen aus den Modulen "Physik und Elektrotechnik", "Übertragungstechnik", "Elektronik" sowie "Digitaltechnik und Rechnersysteme" vorausgesetzt.
Dauer	1

Lernziele

Aufbauend auf den Grundlagen aus den ersten 3 Studiensemestern vermittelt das Modul die wichtigsten Kompetenzen, die für einen systematischen Umgang mit nichtlinearen Schaltungen und Systemen notwendig sind. Am Beispiel von Halbleiterbauelementen und -schaltungen und wird nochmals die Bedeutung der systematischen Betrachtung (Temperaturabhängigkeiten, Parameterstreuungen usw.) verdeutlicht. Wichtigste Schaltungen und Systeme aus dem

Bereich der Halbleiterschaltungstechnik werden vorgestellt und deren Verständnis durch zahlreiche Simulationsbeispiele vertieft. Ein ganz wesentliches Lernziel des Moduls ist außerdem die Herstellung des Bezugs zum den Modulen "Einführung in Digitaltechnik" sowie "Digitaltechnik und Rechnerstrukturen", indem bis dahin meistens nur auf Register-Transfer- oder Gatterebene betrachteten Schaltungen und Systeme nun mit ihren physikalischen Abbildungen in Form von (integrierten) Halbleiterschaltungen versehen werden. Mit dem Absolvieren des Moduls entsteht bei den Studierenden somit ein komplexes Gesamtbild der analogen und digitalen Systeme. Sie werden in die Lage versetzt die Funktionsweise moderner Schaltungen und Systeme von Blockschaltbild bis hin zum Ladungstransport im Halbleiterkristall zu erfassen.

I.1.23.1 Halbleiterschaltungstechnik

Lehrveranstaltung	Halbleiterschaltungstechnik
Dozent(en)	Sergei Sawitzki
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, Tafel

Lernziele

Die Studierenden ...

- erlangen die Kompetenz, analoge und digitale elektronische Systeme, soweit sie für die technische Informatik von Belang sind, zu verstehen und zu entwerfen.
- erlangen der Kompetenz, die für einen erfolgreichen Schaltungsentwurf erforderliche Methodik zu erkennen und einzusetzen.
- bekommen ein Verständnis für realitätsnahe Schaltungsentwicklung unter Einbeziehung des realen Bauteilverhaltens (Toleranzen, Streuungen, Temperaturabhängigkeiten, etc.) sowie Analyse bestehender Schaltungen und Systeme
- bekommen Kenntnissen über Aufbau und Funktionsweise von den wichtigsten Halbleiter-Bauelementen vermittelt
- erwerben Verständnis von Grundzügen der Halbleiterschaltungstechnik
- erkennen Verbindungen zwischen Digitaltechnik und Halbleiterschaltungstechnik als technologischer Grundlage digitaler Schaltungen und Systeme

Inhalt

- Grundlagen der Halbleiterphysik
- Halbleiter-Bauelemente
 - pn-Übergang, Diode
 - Bipolartransistor
 - Feldeffekttransistoren
 - Überblick über sonstige Bauelemente
 - Grundsaltungen, Verstärker
- Halbleiterschaltungstechnik
 - Strom- und Spannungsquellen

- Operationsverstärker
- Transistoren als Schalter, Digitale Schaltungen

Literatur

- Tietze, Ulrich; Schenk, Christoph: Halbleiterschaltungstechnik, 16. Auflage Springer Verlag, 2016
- Stiny, Leonhard: Handbuch aktiver elektronischer Bauelemente, Franzis Verlag 2009
- Hering, Ekbert; Bessler, Klaus; Gutekunst, Jürgen: Elektronik für Ingenieure und Naturwissenschaftler, Springer Verlag 2005
- Horowitz, Paul; Hill, Winfield: Die Hohe Schule der Elektronik. Teil 1: Analogtechnik, 8. Auflage Elektor-Verlag 2006
- Schiffmann, Wolfram; Schmitz, Robert: Technische Informatik, in 3 Bänden. 3. Auflage Springer Verlag, 1996
- Brauer, Harry; Lehmann, Constans; Lindner, Helmut: Taschenbuch der Elektrotechnik und Elektronik, 9. Auflage Hanser Fachbuchverlag 2008
- Taur, Yuan; Ning, Tak H.: Fundamentals of Modern VLSI Devices, 2nd edition, Cambridge University Press 2009
- Ng, Kwok K.: Complete Guide to Semiconductor Devices, 2nd edition, John Wiley & Sons 2002
- Veendrick, Harry: Nanometer CMOS ICs, Springer 2008

I.1.23.2 Übg. Elektronik und Halbleiterschaltungstechnik

Lehrveranstaltung	Übg. Elektronik und Halbleiterschaltungstechnik
Dozent(en)	Timm Bostelmann
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Handout, Softwaredemonstration, studentische Arbeit am Rechner, Tafel

Lernziele

Nachdem Studierende die Veranstaltung besucht haben, können sie

- typische Aufgaben aus dem Stoffumfang der Vorlesungen Elektronik und Halbleiterschaltungstechnik eigenständig lösen.
- Kenndaten elektronischer Bauteile aus Datenblättern entnehmen.

Inhalt

- Passive Netze
 - Filterschaltungen
 - Einschwingvorgänge
 - Stabilisierungsschaltungen
- Verstärkerschaltungen
 - Arbeitspunkteinstellungen
 - Statisches und dynamisches Verhalten
 - Gegenkopplungen
- Operationsverstärkerschaltungen
 - Beschaltungen
 - Anwendungen
 - Stabilität

Literatur

Tietze, Ulrich; Schenk, Christoph:
Halbleiterschaltungstechnik. 14. Auflage
Springer Verlag, 2002

I.1.24 Systemtheorie

B073 Systemtheorie

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B073
Bezeichnung	Systemtheorie
Lehrveranstaltung(en)	B073a Systemtheorie
Verantwortliche(r)	Prof. Dr.-Ing. Carsten Burmeister
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Das Modul legt die theoretischen Grundlagen für das Modul "Regelungstechnik" und kann mit weiteren technisch orientierten Modulen der technischen Informatik und der Ingenieurwissenschaften kombiniert werden.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Die Studierenden sollen grundlegende Kenntnisse der Analysis aus den Bereichen Differentialrechnung, Integralrechnung und Funktionsanalyse besitzen.
Dauer	1

Lernziele

Nach Abschluss des Moduls verfügen die Studierenden über Kenntnisse hinsichtlich der grundlegenden Konzepte verschiedener Integral- und anderer Funktionaltransformationen. Hiermit sind sie in der Lage, Transformationsmethoden in der Konzeption und Dimensionierung von Übertragungssystemen, analogen und digitalen Filtern und Redundanzreduktionsverfahren anzuwenden.

I.1.24.1 Systemtheorie

Lehrveranstaltung	Systemtheorie
Dozent(en)	Carsten Burmeister
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assignm.
Semesterwochenstunden	4
ECTS	5.0
Prüfungsform	Portfolio-Prüfung
Sprache	deutsch/englisch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, studentische Arbeit am Rechner, Tafel

Lernziele

Die Studierenden ...

- kennen zeitkontinuierliche und zeitdiskrete Integraltransformationen und deren Eigenschaften.
- können Integraltransformationen zur Spektralanalyse, zur Lösung linearer Differentialgleichungssysteme und zur allgemeinen Analyse nachrichtentechnischer und regelungstechnischer Systeme anwenden.

Inhalt

- Beschreibung und Analyse von kontinuierlichen Signalen und Systemen im Zeitbereich
 - Signale und Signalarten
 - Systeme und Systemeigenschaften
 - Impulsantwort und Faltung
- Beschreibung und Analyse von kontinuierlichen Signalen und Systemen im Frequenzbereich
 - Fourieranalyse
 - Fourierreihen
 - Fouriertransformation
- Beschreibung und Analyse von kontinuierlichen Signalen und Systemen im Bildbereich
 - Laplacetransformation
 - Systemstabilität
- Beschreibung und Analyse von zeitdiskreten Signalen und Systemen
 - Abtastung / Abtasttheorem
 - Digitale Signalverarbeitung

- z-Transformation
- Diskrete Fouriertransformation und FFT

Literatur

Meyer, M.: Signalverarbeitung. Vieweg und Teubner 2011, 6. Auflage

Oppenheim, A., Willsky, S.: Signals and Systems, 2nd Edition. Pearson 1996, 2. Auflage

Chaparro, L.: Signals and Systems using MATLAB. Academic Press 2018, 3. Auflage

I.1.25 Grundlagen der Computergrafik

B085 Grundlagen der Computergrafik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B085
Bezeichnung	Grundlagen der Computergrafik
Lehrveranstaltung(en)	B085a Grundlagen der Computergrafik B085b Prakt. Grundlagen der Computergrafik
Verantwortliche(r)	Prof. Dr. Christian-Arved Bohn
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Die generative 3D-Computergrafik berührt inhaltlich einige Grundgedanken der 2D Bildbearbeitung. Das Modul "Bildbearbeitung und -analyse" bietet sich daher als Kombination mit diesem Modul an.
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Grundkenntnisse der Mathematik, insbesondere der linearen Algebra und Vektorrechnung
Dauer	1

Lernziele

Lernziele des Moduls sind die erlernten theoretischen Kompetenzen der Vorlesung "Grundlagen der Computergrafik" und die Fähigkeit, diese auch praktisch einsetzen zu können, was im zugehörigen Praktikum vermittelt wird.

I.1.25.1 Grundlagen der Computergrafik

Lehrveranstaltung	Grundlagen der Computergrafik
Dozent(en)	Christian-Arved Bohn
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, Tafel

Lernziele

Studierenden werden Fähigkeiten,

- grundlegende Probleme der generativen Computergrafik einzuordnen und zu klassifizieren und
- entsprechende Lösungsstrategien vorzuschlagen und zu implementieren,

vermittelt.

Inhalt

Die Veranstaltung gibt einen Überblick über wesentliche Aspekte der generativen Computergrafik. Konkret werden die Algorithmen des (a) Raytracings und der Projektion als geometrische Abbildung behandelt und jeweils die Teilaspekte Projektion, Verdeckung und Beleuchtungsrechnung dargestellt. Es werden intensiv Methoden der linearen Algebra besprochen, die geometrische Abbildungen zur Konstruktion von virtuellen Szenen und zur Projektion verwenden. Es folgen praktische Aspekte, die es hier zu beachten gilt und deren hardwarenahe Realisierung (z.B. "Clipping", "Buffer"). Die Technik der Texturierung wird aus mathematischer Sicht behandelt und anhand von praktischen Beispielen erläutert. Einen Einblick in weiterführende Probleme der Computergrafik geben die Grundlagen der globalen Beleuchtungsrechnung ("Rendering Equation").

Literatur

- Hearn, M. P. Baker: Computer Graphics with OpenGL, Pearson Education International, 2004.
- Peter Shirley et al.: Fundamentals of Computer Graphics, A K Peters, 2005.

I.1.25.2 Prakt. Grundlagen der Computergrafik

Lehrveranstaltung	Prakt. Grundlagen der Computergrafik
Dozent(en)	Lars Neumann
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	4
ECTS	3.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, studentische Arbeit am Rechner, Tafel, Tutorien

Lernziele

Die Studierenden

- sind in der Lage, die in der Vorlesung vermittelten theoretischen Hintergründe selbst praktisch mit OpenGL anzuwenden und sicher zu handhaben,
- haben umfangreiche Kenntnisse und praktische Erfahrungen zu den Themen OpenGL, Callback-Prinzip, 2D-Anwendungen, Matrixstacks, 3D-Szenen, Displaylisten, lokale Beleuchtung, Texturierung, Picking, Viewports und Blending,
- haben sich die Grundlagen von Vertex-Arrays, Simulationen und Shadern erarbeitet und
- steigern ihre Teamfähigkeit durch intensive Arbeit in Zweierteams und Kommunikation über auftretende Probleme in der ganzen Gruppe.

Inhalt

OpenGL, affine Transformationen, 2D- und 3D-Anwendungen, lokale Beleuchtungsmodelle, Texturierung, Picking, Viewports, Transparenz, Vertex-Arrays, Simulationen, Shader

Literatur

Skript:

- Vorlesungsskript unter <http://cg.viswiz.de/> => Lehrveranstaltungen => Computergrafik 1
- Weiteres Material unter <http://www.fh-wedel.de/mitarbeiter/ne/praktikum-grundlagen-der-computergrafik-fh/material/>

Online-Quellen:

- The OpenGL Programming Guide - The Redbook (<http://www.glprogramming.com/red/>)
- The OpenGL Reference Manual - The Bluebook (<http://www.glprogramming.com/blue/>)
- NeHe Productions (<http://nehe.gamedev.net/>)

Bücher:

- Computergrafik und OpenGL - Eine systematische Einführung, Dieter Orlamünder / Wilfried Mascolus, Hanser, 2004, ISBN: 3-446-22837-3
- Jetzt lerne ich OpenGL : der einfache Einstieg in die Schnittstellenprogrammierung, Lorenz Burggraf, Markt und Technik, 2003, ISBN: 3-8272-6237-2

I.1.26 Software-Design

B058 Software-Design

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B058
Bezeichnung	Software-Design
Lehrveranstaltung(en)	B058a Software-Design
Verantwortliche(r)	M.Sc. Christian Uhlig
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul setzt unmittelbar auf den Inhalten des Moduls „Programmstrukturen 2“ auf und eignet sich damit als Weiterqualifikation im Anschluss an „Programmstrukturen 2“. Es kann mit fortgeschrittenen Modulen zur Software-Technik kombiniert werden, insbesondere mit dem „Programmierpraktikum“, „Software-Projekt“ und „Fortgeschrittene Objektorientierte Programmierung“.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Vorausgesetzt werden Kenntnisse in imperativen und speziell objektorientierten Programmiersprachen sowie konkret in der Programmiersprache Java. Die Inhalte der Module „Algorithmen und Datenstrukturen“ sowie „Fortgeschrittene Objektorientierte Programmierung“ können gewinnbringend eingeordnet werden, sind aber nicht erforderlich.
Dauer	1

Lernziele

Nach Abschluss des Moduls kennen die Studierenden den Gegenstand und die Bedeutung des Softwareentwurfs im Allgemeinen und des objektorientierten Entwurfs im Speziellen und setzen typische Modellierungskonzepte wie UML für den Entwurf ein. Hierbei beachten sie u.a. Kon-

sistenz, Vollständigkeit, Erweiterbarkeit und Performance des modellierten Softwaresystems in Hinblick auf das entsprechende Fachkonzept.

Die Teilnehmer beherrschen den Einsatz gängiger objektorientierter Entwurfsmuster, erläutern ihre Vor- und Nachteile sowie Alternativen und wählen für gegebene Anwendungsfälle geeignete Entwurfsmuster aus. Insbesondere können sie in diesem Zuge Vererbung und Komposition als Konzepte im objektorientierten Entwurf einsetzen und in ihren Eigenschaften diskutieren.

Ferner leiten die Teilnehmer am Beispiel von Java das Gerüst eines Softwaresystems aus Entwurfsmodellen ab und implementieren die relevanten Operationen entsprechend Entwurf und Fachkonzept.

I.1.26.1 Software-Design

Lehrveranstaltung	Software-Design
Dozent(en)	Christian Uhlig
Hörtermin	4
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	4
ECTS	5.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, interaktive Entwicklung und Diskussion von Modellen, Softwaredemonstration

Lernziele

Die Studierenden ...

- erkennen und erläutern die Einordnung des Entwurfs in den Softwareentwicklungsprozess.
- erkennen und erläutern die Bedeutung der Modellbildung im Softwaredesign.
- differenzieren informelle und formale Entwurfskonzepte.
- erkennen und erläutern die Bedeutung von Verträgen bei Entwurf und Implementierung abstrakter Datentypen.
- erkennen und erläutern das Entwurfparadigma Design by Contract am Beispiel der Programmiersprachen Eiffel und Java.
- differenzieren Vererbung und Komposition als zentrale Konzepte des objektorientierten Entwurfs und wählen problemadäquat aus.
- wenden informelle Notationen und Methoden (OMT, UML, ERD, ...) zur Modellierung eines Softwaresystems an.
- wenden formale Notationen (z.B. Haskell) zur Definition der Datenstrukturen und der Schnittstellen eines Softwaresystems an.
- differenzieren Entwurfsmuster auf Grundlage von Struktur, Motivation und Zielsetzung.
- wenden typische objektorientierte Entwurfsmuster zur Lösung von softwaretechnischen Problemstellungen an.
- nennen und erläutern sprachabhängige und -unabhängige Implementierungsaspekte bei der Anwendung von Entwurfsmustern.
- erkennen und erläutern die Anwendbarkeit und Kombinierbarkeit einzelner Entwurfsmuster.
- differenzieren Flexibilität und Effizienz bei der problembezogenen Auswahl und Anwendung von Entwurfsmustern.

Inhalt

- Einordnung und Bedeutung des Entwurfs im Softwareentwicklungsprozess
- Methoden, Techniken und Werkzeuge im Software-Entwurf
 - informelle Methoden/Notationen: OMT, UML, ERD
 - formale Methoden/Notationen: abstrakte Syntax / Haskell
- Verträge zwischen Softwarekomponenten, Design by Contract
- Objektorientierte Entwurfsmuster
 - Erzeugungsmuster
 - Strukturmuster
 - Verhaltensmuster
- Fallstudien

Literatur

- Uhlig, Christian: Software-Design, Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~uhl/swd.html>
- Balzert, Helmut: Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb, 3. Auflage, Spektrum, 2011
- Balzert, Helmut; Balzert, Heide; Koschke, Rainer; Lämmel, Uwe; Liggesmeyer, Peter; Quante, Jochen: Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering, 3. Auflage, Spektrum, 2009
- Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John: Design Patterns: Entwurfsmuster als Elemente wiederverwendbarer objektorientierter Software, mitp, 2014
- Freeman, Eric; Robson, Elisabeth; Bates, Bert; Sierra, Kathy: Head First Design Patterns, OReilly, 2014
- Rumbaugh, James; Blaha, Michael; Premerlani, William; Eddy, Frederick; Lorensen, William: Objektorientiertes Modellieren und Entwerfen, Hanser, 1994
- Fowler, Martin: UML Distilled: A Brief Guide to the Standard Object Modeling Language, 3. revidierte Auflage, Addison-Wesley, 2003
- Hutton, Graham: Programming in Haskell, 2. Auflage, Cambridge University Press, 2016

I.1.27 Programmierpraktikum

B036 Programmierpraktikum

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B036
Bezeichnung	Programmierpraktikum
Lehrveranstaltung(en)	B036a Programmierpraktikum
Verantwortliche(r)	Dipl.-Ing. (FH) Gerit Kaleck
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul baut auf die im Modul "Programmstrukturen 2" erworbenen Programmierkompetenzen auf. Es bildet die Grundlage für Module von Informatik-Studiengängen, in denen Programmierung von Softwareeinheiten größeren Umfangs und softwaretechnische Aspekte eine Rolle spielen, zum Beispiel die Module "Software-Design", "Software-Projekt" und "Softwarequalität".
Semesterwochenstunden	0
ECTS	5.0
Voraussetzungen	Kenntnis der zentralen Konzepte der Objektorientierten Programmierung und der Programmiersprache Java, Grundkenntnisse in der Benutzung einer Entwicklungsumgebung.
Dauer	1

Lernziele

Die Studierenden verfügen nach Abschluss des Moduls über die Fähigkeit, aus einer textuellen, problemorientiert formulierten Aufgabenstellung die wesentlichen funktionalen Anforderungen an eine zu entwickelnde Software abzuleiten und in Form eines Pflichtenheftes zu dokumentieren. Sie sind zur Konzeption, insbesondere zur angemessenen Modularisierung von Softwaresystemen mittleren Umfangs in der Lage. Sie können die objektorientierte Programmiersprache Java einsetzen, um ein solches Softwaresystem eigenständig objektorientiert zu implementieren. Sie sind fähig, dabei eine moderne Entwicklungsumgebung zu nutzen.

Dabei verfügen sie über das Problembewusstsein im Hinblick auf die benutzungsgerechte Softwaregestaltung und verfügen über Kenntnisse hinsichtlich der Grundregeln von benutzungsgerechter Oberflächengestaltung. Sie verfügen zudem über Fähigkeiten zum angemessenen Einsatz von Interaktionselementen in grafischen Oberflächen bei Aufgabenstellungen mittleren Schwierigkeitsgrades.

Sie verfügen über Basiskenntnisse hinsichtlich der Qualitätssicherung von Software in Form einfacher Teststrategien und können diese einsetzen, um die funktionale Korrektheit und ein ausreichendes Maß an Zuverlässigkeit der Software zu gewährleisten.

Die Studierenden besitzen die Fähigkeit, die Entwicklung einer Software und die Software selbst in einer geeigneten Form zu dokumentieren und zu präsentieren.

I.1.27.1 Programmierpraktikum

Lehrveranstaltung	Programmierpraktikum
Dozent(en)	Gerit Kaleck
Hörtermin	5
Häufigkeit	jedes Semester
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	0
ECTS	5.0
Prüfungsform	Praktikumsbericht / Protokoll
Sprache	deutsch
Lehr- und Medienform(en)	studentische Arbeit am Rechner

Lernziele

Die Studierenden erwerben mit diesem Praktikum die ...

- Fähigkeit zum praktischen Einsatz der Basiskonzepte objektorientierter Programmiersprachen sowie ihrer Umsetzung in Java in einem Softwareprojekt mittleren Umfangs.
- Fähigkeit zur Nutzung einer aktuellen, verbreiteten Entwicklungsumgebung.
- Fähigkeit zur eigenständigen Strukturierung und Realisierung von vollständigen Softwaresystemen mittleren Umfangs ausgehend von einer problemorientierten Aufgabenstellung.
- Fähigkeit zur Softwareentwicklung und -dokumentation.
- Fähigkeit zur Anwendung der Grundregeln benutzungsgerechter Oberflächengestaltung.

Inhalt

- Entwicklung eines vollständigen Softwaresystems mittleren Umfangs in Java ausgehend von einer problemorientierten Aufgabenstellung.
- Strukturierung und Modularisierung des Projektes.
- Eigenständiger Entwurf passender Datenmodelle.
- Benutzungsgerechte Gestaltung der Oberfläche.
- Testen der entstandenen Software und Dokumentation der Tests.
- Erstellen einer Programmdokumentation und eines Benutzerhandbuchs.

Literatur

- ULLENBOOM, Christian:
Java ist auch eine Insel
Rheinwerk Computing, 2020 (15. Auflage)
ISBN-13: 978-3836277372
- EPPLE, Anton:
JavaFX 8: Grundlagen und fortgeschrittene Techniken

dpunkt Verlag, 2015
ISBN-13: 978-3864901690

- ZÖRNER, Stefan:
Softwarearchitekturen dokumentieren und kommunizieren: Entwürfe, Entscheidungen
und Lösungen nachvollziehbar und wirkungsvoll festhalten
Carl Hanser Verlag, 2015 (2. Auflage)
ISBN-13: 978-3446443488

I.1.28 Großintegrierte Systeme

B108 Großintegrierte Systeme

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B108
Bezeichnung	Großintegrierte Systeme
Lehrveranstaltung(en)	B108a Großintegrierte Systeme B108b Workshop Mikroprozessor
Verantwortliche(r)	PD Dr. Dennis Säring
Zuordnung zum Curriculum	Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul "Großintegrierte Systeme" baut auf den in den Modulen "Rechnerstrukturen und Digitaltechnik", "Informationstechnik" und "Algorithmen und Datenstrukturen" erworbenen Kenntnissen und Fähigkeiten auf und vertieft diese im Bezug auf moderne Rechnersysteme mit besonderem Augenmerk auf Mikroprozessor- und Mikrocontroller-Systeme. Um bei den Studierenden ein ganzheitliches Bild von modernen Rechnersystemen zu erzeugen, sollte dieses Modul mit Modulen, die diese auf tieferen Abstraktionsebenen betrachten, kombiniert werden.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Es werden Kenntnisse aus den Modulen "Digitale Systeme", "Informationstechnik" und "Algorithmen und Datenstrukturen" vorausgesetzt.
Dauer	1

Lernziele

Das Ziel des Moduls ist es, den Studierenden einen vertiefenden Einblick in die moderne Mikroprozessor- und Systemtechnik zu liefern. Die Teilnehmer erlangen die Kompetenzen, die zum Begreifen der technologischen Aspekte moderner Rechnersysteme in volle Breite von eingebetteten (auf Mikrocontrollern basierten) bis hin zu großintegrierten Systemen (inklusive Multicore-Prozessoren) notwendig sind, womit eine weitere Vertiefung in Richtung Systemarchitektur erreicht wird. Zusätzlich werden im "Workshop Mikroprozessor" die Grundlagen der hardware-nahen Programmierung vermittelt, womit die Studierenden die Möglichkeiten des Softwarezugangs zur Hardware auf einer tieferen Abstraktionsebene erlernen und anschließend selbständig anwenden können.

I.1.28.1 Großintegrierte Systeme

Lehrveranstaltung	Großintegrierte Systeme
Dozent(en)	Dennis Säring
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	E-Learning

Lernziele

Die Studierenden ...

- besitzen ein grundlegendes Verständnis für die realen, komplexen Implementierungen großintegrierter Funktionsgruppen, wie CPUs, Chipsets und Speichern
- verfügen über elementare physikalische Grundlagen der Schaltungstechnologien und der praktischen Implementierung großintegrierter Funktionselemente
- kennen die Komplexität der Strukturen von einfachen Zellen bis zum kompletten Design aktueller Mainstream-Prozessoren
- haben die Fähigkeit zur Einschätzung alternativer Entwicklungstrends in den Bereichen Speichertechnologie, Chipsatz und Prozessorstruktur

Inhalt

- Technologische Aspekte und Abhängigkeiten bei der Prozessorentwicklung
- Grundkonzepte und Basistechnologien der Schaltkreisfamilien
- Speicher-Technologien
- Prozessorumgebung, Kommunikationskonzepte und Chipsätze
- Betrachtung ausgewählter Monocore-Prozessorstrukturen
- Multicore - Architekturen

Literatur

- Tanenbaum: Computerarchitecture, 4. Auflage, Addison-Wesley
- Hennessy, Patterson: Computer architecture, Morgan Kaufman Publishing
- Diverse aktuelle weblinks für die jeweilige Veranstaltung

I.1.28.2 Workshop Mikroprozessor

Lehrveranstaltung	Workshop Mikroprozessor
Dozent(en)	Jörg Völker
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Projekt
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Sprache	deutsch
Lehr- und Medienform(en)	studentische Arbeit am Rechner

Lernziele

Studierende erlangen die ...

- Fähigkeit zur Erfassung der Struktur und Funktionalität eines realen, industrietypischen Mikroprozessors.
- Fähigkeit zum Erfassen der Struktur und der Eigenschaften einer realen Assemblersprache.
- Kompetenzen zum Schreiben und Austesten von Assemblerprogrammen an realitätsnahen Problemstellungen.

Inhalt

- Einführung in die Struktur vom Mikroprozessoren
- Befehlsstruktur
 - Datenflüsse in der Hardware
 - Befehlstypen
 - Adressierungsarten
 - IRQ-Struktur
- Einführung in die verwendeten Programmier-Tools
- Praktische Übungsaufgaben
 - Beginnend mit einfachen Aufgaben
 - In Folge steigender Schwierigkeitsgrad
- Abschlussaufgabe

Literatur

SCHMITT, Günter:

Mikrocomputertechnik mit den Prozessoren der 68000-Familie. Maschinenorientierte Programmierung. Grundlagen - Schaltungstechnik - Anwendungen.

München: Oldenbourg, 1987

I.1.29 Regelungstechnik

B109 Regelungstechnik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B109
Bezeichnung	Regelungstechnik
Lehrveranstaltung(en)	B109a Regelungstechnik B109b Übg. Simulationssoftware
Verantwortliche(r)	Prof. Dr.-Ing. Carsten Burmeister
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul ist mit dem Modul "Systemtheorie" zu kombinieren.
Semesterwochenstunden	5
ECTS	5.0
Voraussetzungen	Vorausgesetzt werden mathematische Kenntnisse der Analysis und der Linearen Algebra, sowie grundlegende Kenntnisse von Integraltransformationen, wie sie im Modul Systemtheorie vermittelt werden.
Dauer	1

Lernziele

Nach Abschluss des Moduls verfügen die Studierenden über Kenntnisse der grundlegenden Ziele und Verfahren der Regelungstechnik und besitzen die Fähigkeit zur Beurteilung von Stabilität, Dynamik und Genauigkeit offener und geschlossener Regelkreise. Sie besitzen eine genaue Kenntnis über Rechenverfahren im Zeit- bzw. im Bildbereich zur Dimensionierung von Regeleinrichtungen und zum Nachweis der Regelziele. Sie können eindimensionale Systeme analysieren und passende Regelalgorithmen für vorgegebene Aufgabenstellungen entwerfen und sie bezüglich des Erreichens von Regelzielen beurteilen. Praktisch erlangen sie die Fähigkeit, spezielle regelungstechnische Software für Problemlösungen anzuwenden. Sie verfügen über grundlegende Fähigkeiten im Umgang Simulations-Software und können damit Probleme lösen und deren Lösungen visualisieren.

I.1.29.1 Regelungstechnik

Lehrveranstaltung	Regelungstechnik
Dozent(en)	Carsten Burmeister
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assigm.
Semesterwochenstunden	4
ECTS	4.0
Prüfungsform	Klausur + ggf. Bonus
Sprache	deutsch/englisch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, studentische Arbeit am Rechner, Tafel

Lernziele

Die Studierenden haben ...

- Kenntnis der grundlegenden Ziele der Regelungstechnik.
- die Fähigkeit zur Beurteilung von Stabilität, Dynamik und Genauigkeit offener und geschlossener Kreise.
- Kenntnis über Rechenverfahren im Zeit- und Bildbereich zur Dimensionierung von Regeleinrichtungen und zum Nachweis der Regelziele.
- die Fähigkeit, eindimensionale Systeme zu analysieren, passende Regelalgorithmen für vorgegebene Aufgabenstellungen zu entwerfen und sie bezüglich des Erreichens von Regelzielen zu beurteilen.

Inhalt

- Einführung in die Regelungstechnik
 - Anwendungsbeispiele
 - Regelziele Genauigkeit, Dynamik, Stabilität
- Grundlagen der Systemtheorie
 - Strukturdiagramme
 - Linearisierung und Normierung
 - Lösung linearer Differentialgleichungen durch Laplace-Transformation
- Systemfunktionen
 - Einfache Funktionen: P-, I-, D-, Tt-Glieder
 - Zusammengesetzte Funktionen: PT1-, DT1-, PT2-Glieder
 - Lineare Regelalgorithmen: P-, I-, PI-, PID-Regler
 - Unstetige Regler: Zweipunkt-, Dreipunktregler ohne und mit Rückkopplungen

- Analyse von Regelkreisen
 - Übertragungsfunktionen offener und geschlossener Kreise
 - Regelung einfacher Kreise
 - Beurteilung von Regelzielen
- Stabilität
 - Wurzelortsverfahren
 - Das Nyquist-Kriterium
 - Frequenzkennlinienverfahren
- Empirische Verfahren
 - Ziegler-Nichols Verfahren
 - Wendetangentenverfahren nach Schwarze

Literatur

- Lunze, J.: Regelungstechnik 1 - Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen, Springer Verlag, 2010
- Zacher, S., Reuter, M.: Regelungstechnik für Ingenieure - Analyse, Simulation und Entwurf von Regelkreisen, Springer Verlag 2014

I.1.29.2 Übg. Simulationssoftware

Lehrveranstaltung	Übg. Simulationssoftware
Dozent(en)	Carsten Burmeister
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assigm.
Semesterwochenstunden	1
ECTS	1.0
Prüfungsform	Abnahme
Sprache	deutsch/englisch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden haben ...

- die Fähigkeiten im Umgang mit Numerik- und Simulations-Software.
- Kenntnisse über die Arbeitsweise und Umfang industrieeüblicher Numerik- und Simulationssoftware.
- Fähigkeiten der typischen Programmierung in den jeweiligen Script-Sprachen.
- die Fähigkeit mit den Methoden und Techniken zur Visualisierung numerischer Probleme umzugehen.

Inhalt

- Einführung in MATLAB / Octave + Simulink
- Entwicklung von Ortskurven zum Umgang mit komplexen Zahlen
- Funktionen und Strukturelemente an Messreihen mit Modulosprüngen
- Grafische Lösungen von Differentialgleichungen
- Umschreibung von Differentialgleichungen höherer Ordnung in Matrixform
- Visualisierung numerischer Lösungen
- Lösung nichtlinearer Gleichungen
- Lösung linearer Systeme
- Programmieren einfacher digitaler Filter

Literatur

- BEUCHER, Ottmar: MATLAB und Simulink: Grundlegende Einführung für Studenten und Ingenieure in der Praxis. 4. Aufl., Pearson Studium, 2008
- OKORO, Ogbonnaya Inya; CHIKUNI, Edward: The Essential MATLAB & Simulink: For Engineers and Scientists. Juta Legal and Academic Publishers, 2009

I.1.30 Seminar Technische Informatik

B148 Seminar Technische Informatik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B148
Bezeichnung	Seminar Technische Informatik
Lehrveranstaltung(en)	B148a Seminar Technische Informatik
Verantwortliche(r)	jeweiliger Dozent
Zuordnung zum Curriculum	Technische Informatik (Bachelor)
Verwendbarkeit	Die im Modul "Seminar Technische Informatik" erworbenen Kompetenzen stellen die Grundlage für die Bachelor-Thesis dar. Die bereits erworbenen Kenntnisse und Fähigkeiten (unter anderem zum Beispiel aus dem Modul "Soft Skills") werden in inhaltlicher, formaler und methodischer Hinsicht zielgerichtet weiterentwickelt. Somit wird dem Studiengangsziel des Studiengangs "Bachelor Technische Informatik" Rechnung getragen, soziale und fachliche Kompetenzen zu vertiefen und in fachübergreifenden Teams sinnvoll einbringen zu können.
Semesterwochenstunden	2
ECTS	5.0
Voraussetzungen	Fachliche und persönliche Kompetenzen der zurückliegenden Semester, insbesondere themenabhängig fachverwandte Module und "Soft Skills".
Dauer	1

Lernziele

Ziel des Moduls Seminar ist es, die Fähigkeit zu erwerben, sich eigenständig in ein anspruchsvolles Thema aus dem Bereich der technischen Informatik einzuarbeiten zu können und dieses geeignet, sowohl im Rahmen eines Vortrags als auch in Form einer schriftlichen Ausarbeitung darstellen zu können. Als Themen werden dabei aktuelle Entwicklungen in der technischen Informatik aber auch grundlegende Themen der Mikroelektronik und Halbleiterschaltungstechnik gewählt. Studierende erlernen, gezielte Literaturrecherchen, insbesondere unter Berücksichtigung der Quellen des Internets, durchzuführen. Sie entwickeln und optimieren ihre Fähigkeiten zur Präsentation des Themas in freien Vorträgen, beim Umgang mit Präsentationsmedien und durch die offene Diskussion wissenschaftlicher Themen in der Gruppe. Die Erstellung einer stilistisch und fachlich ansprechenden Ausarbeitung dient der Vorbereitung auf die Bachelor-Arbeit.

I.1.30.1 Seminar Technische Informatik

Lehrveranstaltung	Seminar Technische Informatik
Dozent(en)	jeweiliger Dozent
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Seminar
Semesterwochenstunden	2
ECTS	5.0
Prüfungsform	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, interaktive Entwicklung und Diskussion von Modellen, Online-Aufbereitung, Software-demonstration, studentische Arbeit am Rechner, Tafel

Lernziele

Seminararbeiten dienen insbesondere dem Erlernen von Fertigkeiten zum Erstellen der Bachelor-Thesis. Wesentlich ist die eigenständige Erarbeitung und Darlegung der Inhalte zu einem vorgegebenen Thema unter Einhaltung der Formalia. Die Ausarbeitung soll das Interesse an einer eigenständigen Befassung mit Inhalten aus dem Themengebiet und den Einstieg in die zugehörige wissenschaftliche Fachliteratur und Methodik fördern und anregen. Schließlich ist die obligatorische Präsentation der Ergebnisse ebenfalls Aufgabe innerhalb des Seminars.

Nach erfolgreicher Teilnahme können sie ...

- wissenschaftliche Themen angemessen strukturieren.
- eine eigenständige Zielsetzung erarbeiten und umsetzen.
- Inhalte recherchieren und übersichtlich aufbereiten.
- formale Kriterien sicher beachten und anwenden.
- eine schriftliche Ausarbeitung größeren Umfangs erstellen.
- kontroverse Lehrmeinungen und aktuelle Trends zu einem Thema herausarbeiten.
- ihre Ergebnisse in angemessener Form vortragen und mit den Seminarteilnehmern diskutieren.

Inhalt

Fachvorträge mit anschließender Gruppendiskussion.

Literatur

Recherche nach aufgabenbezogener Literatur, teilweise aufgabenspezifische Vorgabe einzelner Literaturquellen.

Empfehlungen zur Einführung in das wissenschaftliche Arbeiten

- Axel Bänsch, Dorothea Alewell, Wissenschaftliches Arbeiten, 11. Aufl., München [u.a.]: Oldenbourg 2013.
- Werner Heister, Dagmar Weßler-Poßberg, Studieren mit Erfolg: Wissenschaftliches Arbeiten für Wirtschaftswissenschaftler, 2., überarbeitete Auflage, Stuttgart: Schäffer-Poeschel 2011.
- Jens Hiller, Arbeitstechniken und wissenschaftliches Arbeiten, Herne: Kiehl 2017.
- Walter Krämer, Wie schreibe ich eine Seminar- oder Examensarbeit. 3., überarbeitete und aktualisierte Aufl., Frankfurt: Campus 2009.
- Lydia Prexl, Mit digitalen Quellen arbeiten. Richtig zitieren aus Datenbanken, E-Books, YouTube & Co., 2., aktualisierte und erweiterte Aufl., Paderborn: Ferdinand Schöningh (UTB) 2016.
- Manuel René Theisen, Wissenschaftliches Arbeiten: Technik - Methodik Form, 15. Aufl., München: Vahlen 2011.

I.1.31 Einführung in die Betriebswirtschaft

B034 Einführung in die Betriebswirtschaft

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B034
Bezeichnung	Einführung in die Betriebswirtschaft
Lehrveranstaltung(en)	B034a Einführung in die Betriebswirtschaft
Verantwortliche(r)	M.Sc. Fikret Koyuncu
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Das Modul "Einführung in die Betriebswirtschaft" ist ein Einführungsmodul. Die erworbenen Kompetenzen stellen wesentliche Grundlagen für eine Vielzahl weiterer Module dar, wie zum Beispiel "Operatives Produktionsmanagement", "Business Planning" oder "Controlling & Unternehmensführung".
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Keine
Dauer	1

Lernziele

Nach erfolgreicher Teilnahme an diesem Modul können die Studierenden die Bedeutung von betriebswirtschaftlichen Denkweisen und Methoden für die moderne Unternehmensführung abschätzen. Sie kennen grundlegende Fragestellungen und Methoden zu deren Bearbeitung aus dem Bereich der Allgemeinen Betriebswirtschaftslehre. Die Studierenden sind befähigt, ausgewählte Aufgaben, wie sie sich in der Praxis des Unternehmens ergeben, unter Anwendung betriebswirtschaftlicher Methoden zu lösen.

Die Studierenden können wechselseitige Abhängigkeit zwischen den Aufgaben aus den Bereichen der Betriebswirtschaftslehre, den Ingenieurwissenschaften und der Informatik identifizieren und benennen.

I.1.31.1 Einführung in die Betriebswirtschaft

Lehrveranstaltung	Einführung in die Betriebswirtschaft
Dozent(en)	Fikret Koyuncu
Hörtermin	5
Häufigkeit	jedes Semester
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assigm.
Semesterwochenstunden	4
ECTS	5.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Tafel

Lernziele

Die Studierenden können ...

- das Erfahrungs- und Erkenntnisobjekt der Betriebswirtschaftslehre benennen,
- die Begriffe Wirtschaften und Ökonomisches Prinzip erklären sowie eine Break-Even-Analyse durchführen,
- Unternehmensziele aufzählen; die Aufgaben der Zielbildung erläutern sowie den Zielbildungsprozess wiedergeben,
- ausgewählte Kennzahlen ausrechnen,
- Ziele der Unternehmensführung erläutern, Führungsebenen voneinander abgrenzen, den Führungsprozess beschreiben sowie ausgewählte Führungsstile erläutern und -prinzipien erklären,
- Standortfaktoren identifizieren und Modelle zur Standortbewertung einsetzen,
- die Ziele der Materialwirtschaft wiedergeben und durch Anwendung von Methoden materialwirtschaftliche Analysen durchführen und Handlungsanweisungen ableiten,
- ausgewählte Erzeugnisstrukturdarstellungen für gegebene Problemstellungen erstellen und mit programmorientierten Verfahren die Materialbedarfsplanung durchführen,
- mit ausgewählten Verfahren die optimale Bestellmenge bestimmen,
- den Input, Throughput und Output von Produktionsprozessen beschreiben,
- das optimale Produktionsprogramm für ausgewählte Fälle ermitteln,
- ausgewählte Aufgaben der Produktionsprozessplanung ausführen,
- die Ziele des Marketings nennen, Methoden zur Ableitung der Marketing-Strategie beschreiben und anwenden sowie die Instrumente des Marketing-Mix erläutern,
- Investitionsarten voneinander abgrenzen; den Investitionsprozess beschreiben und die Aufgabe der Investitionskontrolle skizzieren sowie die Vorteilhaftigkeit einer Investition mittels Methoden beurteilen,
- die Ziele und Aufgaben der Finanzwirtschaft nennen; die Finanzierung aus Abschreibungen erläutern sowie den Financial-Leverage-Effekt an einem Beispiel demonstrieren,

- die Bedeutung informationstechnischer Systeme zur Bewältigung betriebswirtschaftlicher Aufgaben erläutern.

Inhalt

Die Studierenden erlernen Grundtatbestände der Betriebswirtschaftslehre, beginnend vom Erfahrungs- und Erkenntnisobjekt dieser wissenschaftlichen Disziplin, über zu fällende konstitutive Entscheidungen, bis hin zu den diversen betriebswirtschaftlichen Funktionen innerhalb eines Betriebes.

Letztere stehen im Mittelpunkt der Veranstaltung. Die theoretischen Inhalte werden durch Praxisbeispiele untersetzt.

Durch zahlreiche Übungen wird das Verständnis für die betriebswirtschaftlichen Prozesse und deren Zusammenhänge gefestigt sowie das eigenständige Arbeiten gefördert.

Inhalte der Veranstaltung sind im Einzelnen:

- Betriebswirtschaftslehre als wissenschaftliche Disziplin
- Grundlagen der Betriebswirtschaftslehre
- Standortwahl
- Unternehmensführung
- Materialwirtschaft
- Produktionswirtschaft
- Marketing & Absatz
- Investition & Finanzierung
- Umfangreiche Übungen zu verschiedenen Vorlesungsteilen

Literatur

- BECKER, Hans Paul: Investition und Finanzierung. 7. akt. Aufl. Wiesbaden: Gabler, 2016
- BERNECKER, Michael: Grundlagen der Betriebswirtschaftslehre. 4. Aufl. Köln: Johanna, 2011.
- BLOHM, Hans; LÜDER, Klaus; SCHÄFER, Christina: Investition. 10. akt. Aufl. München: Vahlen, 2012
- DÄUMLER, Klaus-Dieter; GRABE, Jürgen: Grundlagen der Investitions- und Wirtschaftlichkeitsrechnung. 12. vollst. überarbeitete Aufl. Berlin; Herne: Neue Wirtschafts-Briefe, 2007
- JUNG, Hans: Allgemeine Betriebswirtschaftslehre. akt. 13. Aufl. München: Oldenbourg, 2016
- SCHIERENBECK, Henner; WÖHLE, Claudia: Grundzüge der Betriebswirtschaftslehre. 18. überarb. Aufl. München: Oldenbourg, 2012
- SPECHT, Olaf; SCHMITT, Ulrich: Betriebswirtschaftslehre für Ingenieure + Informatiker. 5. Aufl. München; Wien: Oldenbourg, 2000

- THOMMEN, Jean-Paul; ACHLEITNER, Ann-Kristin: Allgemeine Betriebswirtschaftslehre. 7. vollst. überarb. Aufl. Wiesbaden: Gabler, 2012
- VAHS, Dietmar; SCHÄFER-KUNZ, Jan: Einführung in die Betriebswirtschaftslehre. 7. überarb. Aufl. Stuttgart: Schäffer-Poeschel, 2015.
- WEBER, Wolfgang; KABST, Rüdiger: Einführung in die Betriebswirtschaftslehre. 9. akt. u. überarb. Aufl. Wiesbaden: Gabler, 2014
- WÖHE, Günter; DÖRING, Ulrich: Einführung in die Allgemeine Betriebswirtschaftslehre. 26. überarbeitete und aktualisierte Aufl. München: Vahlen, 2016

I.1.32 Einführung in Datenbanken

B052 Einführung in Datenbanken

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B052
Bezeichnung	Einführung in Datenbanken
Lehrveranstaltung(en)	B052a Einführung in Datenbanken B052b Übg. Einführung in Datenbanken
Verantwortliche(r)	Dr. Michael Predeschly
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Das Modul komplementiert Einführungen in die Programmierung ("Einführung in die Programmierung", "Programmstrukturen 1") in allen Studiengängen. Es ist mit den fortgeschrittenen Modulen "Datenbanktheorie und -implementierung" (Bachelor) und "Konzepte der Datenbanktechnologie" (Master) kombinierbar. Das Modul sollte in allen Studiengängen verwendet werden, in denen Datenhaltung wesentlich ist.
Semesterwochenstunden	3
ECTS	5.0
Voraussetzungen	Vorausgesetzt werden Grundkenntnisse in Programmierung und die Fähigkeit, abstrakt zu denken.
Dauer	1
Lernziele	Nachdem Studierende die Veranstaltungen des Moduls besucht haben, haben sie die Fähigkeit, eine relationale Datenbank unter Nutzung von SQL abzufragen, einzurichten und die betrieb-

liche Informationsverarbeitung mittels relationaler Datenbanksysteme unter Nutzung von SQL zu planen und durchzuführen. Zudem haben sie die Fähigkeit, selbständig einen Datenbankentwurfsprozess unter Verwendung des Entity-Relationship-Datenmodells und des relationalen Datenmodells durchzuführen.

I.1.32.1 Einführung in Datenbanken

Lehrveranstaltung	Einführung in Datenbanken
Dozent(en)	Michael Predeschly
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, E-Learning, Online-Aufbereitung, Tafel

Lernziele

Die Studierenden ...

- beherrschen die Grundlagen der relationalen Datenbanktechnologie;
- erlangen die Fähigkeit, selbstständig einen Datenbankentwurfsprozess zu planen, eine relationale Datenbank unter Nutzung von SQL einzurichten und die Informationsverarbeitung mittels relationaler Datenbanksysteme unter Nutzung von SQL durchzuführen;
- erlangen die Fähigkeit, mit einem Entwurfstool einen Datenbankentwurfsprozess durchzuführen und mittels SQL selbstständig Anfragen an ein Datenbanksystem zu stellen.

Inhalt

- Einführung in die Datenbanktechnologie
- Datenbanksprache SQL - Einführung
- Datenbank-Abfrage mit SQL
- Datenbanksprache SQL - Einrichten der Datenbank
- Das Entity-Relationship-Datenmodell
- Das Relationale Datenmodell
 - Relationenschemata und Datenabhängigkeiten
 - Relationale Datenbanken
 - Normalformen
- Datenbank - Lebenszyklus

Literatur

- Elmasri, Ramez; Navathe, Shamkant B.: Grundlagen von Datenbanksystemen. 3. Aufl. München: Pearson -Verlag, 2009.
- Meier, Andreas: Relationale Datenbanken Leitfaden für die Praxis. Berlin: Springer-Verlag, 2004.

- Vetter, Max: Aufbau betrieblicher Informationssysteme mittels konzeptioneller Datenmodellierung. 8. Aufl. Stuttgart: Vieweg-Teubner, 1998.
- Vossen, Gottfried: Datenmodelle, Datenbanksprachen und Datenbank-Management-Systeme. 5. Aufl. Oldenbourg: Oldenbourg-Wissenschaftsverlag, 2008.

I.1.32.2 Übg. Einführung in Datenbanken

Lehrveranstaltung	Übg. Einführung in Datenbanken
Dozent(en)	Marco Pawlowski
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	1
ECTS	2.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden ...

- besitzen die Fähigkeit, ein Datenbanksystem mit SQL zu befragen und in nicht-triviale textuelle Anfrageanforderungen in SQL zu überführen.
- haben grundlegende Kenntnisse über die Ausführung der von ihnen gestellten Anfragen.
- haben die Kompetenz, ein Datenbankentwurfswerkzeug grundlegend zu bedienen.

Inhalt

Vorlesungsbegleitende praktische Übungen in SQL und zum Datenbankentwurf

Literatur

Vorlesungsunterlagen

I.1.33 Einführung in die Robotik

B107 Einführung in die Robotik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B107
Bezeichnung	Einführung in die Robotik
Lehrveranstaltung(en)	B107a Einführung in die Robotik B107b Prakt. Robotik
Verantwortliche(r)	Prof. Dr. Ulrich Hoffmann
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul lässt sich sinnvoll mit dem Modulen "Bildbearbeitung und -analyse" und "Projekt Eingebettete Systeme" kombinieren. Es wendet Inhalte der Module "Elektronik", "Halbleiterschaltungstechnik" und "Systemnahe Programmierung" praktisch an und kann damit gut in technischen Studiengängen verwendet werden. In einem konsekutiven Studiengang kann das Modul als Grundlage für das Master-Modul "Robotics" dienen.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Voraussetzung sind grundlegende Kenntnisse der Informationsverarbeitung und der Programmier Techniken.
Dauer	1

Lernziele

Ausgehend von den Entwicklungstendenzen im Bereich der Flexiblen Fertigungstechnik wird die grundlegende Kompetenz für das Verständnis der Funktionsweisen und Einsatzschwerpunkten von Industrierobotern geschaffen.

Neben der Betrachtung der technischen Grundlagen liegt ein weiterer Schwerpunkt darin, die Kompetenz zu entwickeln, die Möglichkeiten der Verbindung von Robotern mit "intelligenten" Sensoren zu durchdringen. Die Erkennung und Einschätzung der Eigenschaften optischer Sensorsysteme spielt dabei eine zentrale Rolle.

Konzepte der Offline-Programmierung von Industrierobotern werden an verschiedenen Beispielen erkennbar.

Zudem erwerben Studierende das Verständnis der aktuellen Entwicklungstendenzen zur Erhöhung der Selbständigkeit bei Robotern.

Das Praktikum Robotik vertieft die in der Vorlesung vermittelten Kompetenzen im Rahmen eigener Erfahrungen. Die Studierenden lernen hierbei die Funktion und Nutzung industrieller Roboterprogrammiersysteme sowie die Lösung typischer Problemstellungen mittels grundlegender Algorithmen kennen.

Die Gewinnung von Praxiskompetenz erfolgt mittels softwaretechnischer Umsetzung, exemplarischer Aufgabenstellungen, sowie einer schriftlichen Dokumentation.

I.1.33.1 Einführung in die Robotik

Lehrveranstaltung	Einführung in die Robotik
Dozent(en)	Ulrich Hoffmann
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	

Lernziele

Die Studierenden ...

- besitzen fundierte Kenntnisse der technischen Grundlagen der Robotik.
- besitzen ausgehend von den Entwicklungstendenzen im Bereich der Flexiblen Fertigungstechnik die grundlegende Kompetenz für das Verständnis für Funktionsweisen und Einsatzschwerpunkte von Industrierobotern.
- verfügen über die Kompetenz, die Möglichkeiten der Verbindung von Robotern mit "intelligenten" Sensoren zu durchdringen, insbesondere die Erkennung und Einschätzung der Eigenschaften optischer Sensorsysteme.
- können verschiedene Konzepte der Offline-Programmierung von Industrierobotern identifizieren.
- verstehen aktuelle Entwicklungstendenzen zur Erhöhung der Selbständigkeit bei Robotern.

Inhalt

- Strukturen der Fertigungstechnik
- Flexible Fertigungszellen
- Industrierobotern
- Strukturen und Aufbau von Robotern
- Kinematik
- Antriebe
- Effektoren
- Steuerstrategien
- Koordinatentransformationen
- Punkt-zu-Punkt-Steuerung
- Steuerung mit Interpolation

- Mensch-Maschine-Kommunikation
- Roboter-Programmiersysteme
- Roboter-Sprachen
- Intelligente Sensorik
- Integration von Optischen Sensoren

Literatur

- McKerrow: Introduction to Robotics, Addison Wesley, 1990
- Wirth: Flexible Fertigungssysteme, Hüthig-Verlag
- Vukobratovic: Introduction to Robotics, Springer, 1995
- Blume, Dillmann: Frei Programmierbare Roboter, Vogel Verlag
- Blume, Jakob: Programmiersprachen für Industrieroboter, Vogel Verlag, 1985

I.1.33.2 Prakt. Robotik

Lehrveranstaltung	Prakt. Robotik
Dozent(en)	Ulrich Hoffmann
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Praktikumsbericht / Protokoll
Sprache	deutsch
Lehr- und Medienform(en)	

Lernziele

Nach Bearbeitung des Praktikums sind die Studierenden in der Lage ...

- Lerninhalte der Vorlesung im Rahmen eigener Erfahrungen zu vertiefen.
- Ein gegebenes Roboterprogrammiersystem zu nutzen.
- Techniken zum Sammeln, Glätten und Bewerten von Sensordaten anzuwenden.
- Typische Problemstellungen mittels grundlegender Algorithmen zu lösen.
- Eigene Ideen und Lösungsansätze zu implementieren.
- Versuchsergebnisse in einer schriftlichen Dokumentation zu präsentieren.

Inhalt

Anhand eines Projekts werden die Inhalte aus der Vorlesung praktisch umgesetzt. Die konkreten Zielsetzungen werden jedes Jahr angepasst. Schwerpunkt liegt bei den Grundlagen zur Programmierung mobiler Roboter bis hin zu einfachem autonomen Fahren. Der Charakter des Praktikums liegt beim praktischen betreuten Umsetzen von Verfahren, welche zuvor in der Vorlesung präsentiert worden sind.

Literatur

- Hertzberg, J: "Mobile Roboter: Eine Einführung aus Sicht der Informatik", eXamen.press, 2012
- Prat: "Sensordatenfusion und Bildverarbeitung zur Objekt- und Gefahrenerkennung", 2010

I.1.34 Anwendungen der Künstlichen Intelligenz

B095 Anwendungen der Künstlichen Intelligenz

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B095
Bezeichnung	Anwendungen der Künstlichen Intelligenz
Lehrveranstaltung(en)	B095a Anwendungen der Künstlichen Intelligenz
Verantwortliche(r)	Prof. Dr. Sebastian Iwanowski
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Das Modul liefert praktische Anwendungen aus verschiedenen Bereichen der teilnehmenden Studiengänge. Es gibt Ideen für das Praktikum und die anschließende Bachelor-Thesis. Es liefert Grundlagen, die zur Aufnahme eines Masterstudiums motivieren.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Erwartet werden Kenntnisse der Diskreten Mathematik sowie gute Programmierkenntnisse. Die Teilnehmer sollten bereits größere Programme geschrieben haben und Problemstellungen aus der Praxis kennengelernt haben (mindestens im Rahmen angewandter Vorlesungen). Vertrautheit mit objektorientierter Programmierung ist von Vorteil.
Dauer	1

Lernziele

Nach Abschluss des Moduls kennen die Studierenden die grundsätzlichen Ziele der Künstlichen Intelligenz und deren Anwendungsbezug. Die Studierenden kennen komplexe Anwendungsbeispiele und sind in der Lage, elementare Techniken der Künstlichen Intelligenz in Implementierungen anzuwenden. Hierfür verfügen sie über eine grundlegende Kenntnis wichtiger Basistechnologien der Künstlichen Intelligenz.

I.1.34.1 Anwendungen der Künstlichen Intelligenz

Lehrveranstaltung	Anwendungen der Künstlichen Intelligenz
Dozent(en)	Sebastian Iwanowski
Hörtermin	5
Häufigkeit	jährlich
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assig.
Semesterwochenstunden	4
ECTS	5.0
Prüfungsform	Klausur + ggf. Bonus
Sprache	deutsch/englisch
Lehr- und Medienform(en)	

Lernziele

Nach Abschluss der Veranstaltung besitzen die Studierenden folgende Kompetenzen:

- Kenntnis und Interesse für die grundsätzlichen Ziele der Künstlichen Intelligenz.
- Kenntnis der Basistechnologien der Künstlichen Intelligenz.
- Fähigkeit, elementare Techniken der Künstlichen Intelligenz in Implementierungen anzuwenden.
- Kenntnis verschiedener komplexer Anwendungsbeispiele.

Inhalt

- Einführung
 - Definition und Ziele der KI
 - Überblick über die Basistechnologien der KI
 - Auswahl von Anwendungsbeispielen
- Basistechnologien
 - Wissensbasierte Systeme mit den Ausprägungen Regelbasierte Systeme, Modellbasierte Systeme und Fallbasierte Systeme
 - Machine Learning
 - Suchstrategien
 - Schwarmintelligenz
 - Grundlagen von semantischen Netzwerken
- Anwendungen
 - Verkehrsinformation und -navigation
 - Logistische Fragestellungen
 - Technische Diagnose

- Bilderkennung
- Spiele

Literatur

- Marco Dorigo / Thomas Stützle:
Ant Colony Optimization,
MIT Press 2004, ISBN 0-262-04219-3
- Goodfellow, Ian, Yoshua Bengio und Aaron Courville: Deep Learning. <http://www.deeplearningbook.org>. MIT Press, 2016. ISBN: 978-0-262-03561-3
- Ute Schmid / Günter Görz / Josef Schneeberger:
Handbuch der Künstlichen Intelligenz,
Oldenbourg 2013 (5. Auflage), ISBN 978-3-486-71307-7
- Stuart Russell / Peter Norvig:
Künstliche Intelligenz: Ein moderner Ansatz,
Pearson Studium 2012 (3. Auflage), ISBN 978-3-86894-098-5
- Liyang Yu : A Developer's Guide to the Semantic Web , Springer 2011, ISBN 978-3-642-15969-5

I.1.35 Netzwerk- und Messtechnik

B111 Netzwerk- und Messtechnik

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B111
Bezeichnung	Netzwerk- und Messtechnik
Lehrveranstaltung(en)	B111a Workshop Messtechnik B111b Workshop Rechnernetze
Verantwortliche(r)	Prof. Dr.-Ing. Carsten Burmeister
Zuordnung zum Curriculum	IT-Ingenieurwesen (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	In dem Modul werden theoretische Kenntnisse der Rechnernetze und der elektrotechnischen Grundlagen vertieft. Daher ist es mit den Modulen "Rechnernetze", "Elektrotechnik" und "Physik" oder "Grundlagen der Elektrotechnik" zu kombinieren.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Die Studierenden sollen grundlegende Kenntnisse der Elektrotechnik, der Physik, der Übertragungstechnik, der Ingenieurmathematik sowie der Informationstechnik besitzen.
Dauer	1

Lernziele

Nach Abschluss des Moduls verfügen die Studierenden über allgemeines Grundlagenwissen zur Funktionsweise meist digitaler Übertragungssysteme und tieferes Wissen hinsichtlich technischer Aspekte zum Aufbau und Betrieb moderner Unternehmensnetze. Durch den Workshop Netzwerke verfügen sie über Fähigkeiten insbesondere zum praktischen Aufbau einer Infrastruktur nach aktuellen Standards und Normen. Mittels des Workshop Messtechnik besitzen die Studierenden Kenntnisse hinsichtlich dem Einsatz moderner messtechnischer Methoden (Hardware und Software) zur Extraktion von Systemeigenschaften in jeweils einem konkreten Einsatzbereich der Übertragungstechnik. Sie sind in der Lage, derartiges Wissen sich selbstständig zu erarbeiten.

I.1.35.1 Workshop Messtechnik

Lehrveranstaltung	Workshop Messtechnik
Dozent(en)	Carsten Burmeister
Hörtermin	5
Häufigkeit	jedes Semester
Lehrform	Workshop
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Sprache	deutsch/englisch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, interaktive Entwicklung und Diskussion von Modellen, Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden haben die Fähigkeit zum Einsatz moderner messtechnischer Methoden (Hardware und Software) zur Extraktion und Nutzung von Systemeigenschaften in jeweils einem konkreten Einsatzbereich.

Inhalt

- Grundlagen messtechnischer Methoden
- Messfehleranalyse
- Modellbildung
- Fehlerminimierung
- Selbständiger Einsatz messtechnischer Verfahren in einem konkreten Projekt

Literatur

- Best, R.: Digitale Signalverarbeitung und Simulation, Bd. 1 & 2, AT Verlag, 1989
- Isermann, R.: Identifikation dynamischer Systeme, Bd. 1 & 2, Springer, 1988
- Introduction to Random Signals and Applied Kalman Filtering, John Wiley & Sons, 1997
- weitere projektbezogene Literatur

I.1.35.2 Workshop Rechnernetze

Lehrveranstaltung	Workshop Rechnernetze
Dozent(en)	Ilja Kaleck
Hörtermin	5
Häufigkeit	jedes Semester
Lehrform	Workshop
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Sprache	deutsch
Lehr- und Medienform(en)	E-Learning, Online-Aufbereitung, Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden erlangen ...

- erweitertes Wissen über den technischen Aufbau und Betrieb aktueller Unternehmensnetze.
 - Kenntnisse über praxisrelevante Techniken zum Aufbau moderner Unternehmensnetze.
 - die Fähigkeit zum Aufbau flexibler Netzstrukturen unter Nutzung Virtueller LANs (VLANs).
 - die Fähigkeit zur Konfiguration und zum Management aktueller Layer-2 Komponenten.
 - die Grundfertigkeit zur Konfiguration typischer Routersysteme auf Basis des Cisco Internetwork Operating System (IOS).
- Kenntnisse zur effizienten Übertragung vornehmlich multimedialer Daten (Audio, Video) in Datennetzen bzw. dem Internet.
 - Grundkenntnisse über aktuelle Routingprotokolle für das IPv4-Multicasting.
 - Kenntnisse zur IP-Multicast Konfiguration vornehmlich auf Cisco-Routern.
- Kenntnisse über aktuelle Normen zum Aufbau einer strukturierten Gebäudeverkabelung.
 - die Fähigkeit zum praktischen Aufbau einer normgerechten, passiven Übertragungstrecke (typ. CAT-5 / 5e / Class-D) und Verständnis zugehöriger Streckenparameter.
 - die Fähigkeit zur Durchführung normierter Messverfahren an passiven Übertragungstrecken in Datennetzen und Deutung zugehöriger Messergebnisse.
- die Fähigkeit zur Einarbeitung in einen selbst gewählten aktuellen Themenkomplex im Umfeld von Rechnernetzen. Sie erläutern und stellen selbst erarbeitete Projektergebnisse vor.

Inhalt

- Einrichtung einer eigenen PC-Arbeitsstation (Workstation bzw. Server)
 - Anschluss an bestehende Labornetze (mehrere) per VLAN-Technik
 - Einrichtung weiterer virtueller Arbeitsstation (VMs) mit dedizierter VLAN-Ankopplung
- Aufbau einer eigenen passiven LAN-Infrastruktur (Verkabelung)
 - Aufbau einer passiven Übertragungsstrecke und Messung elementarer Signalparameter mit einem Kabelscanner
 - Untersuchung der Eigenschaften verschiedener Übertragungsstrecken; inkl. Fehleranalyse
- Aufbau einer eigenen lokalen LAN- bzw. Layer-2/3 Netzinfrastruktur im 19-Rack
 - Anschluss des Aufbaus an das Labornetz
 - Konfiguration eines managebaren Layer-2 Switches (Cisco SG-Modellreihe)
 - Grundlegendes Switch-Management per Kommandozeile (Cisco-IOS konformer CLI)
 - VLAN-Einrichtung auf dem Switch sowie Konfigurationsmanagement (per TFTP)
 - Einsatz der Port-Spiegelung zur Traffic-Analyse per LAN-Analyzer
 - Aktivierung und Konfiguration des Layer-3 Switching (IPv4/IPv6-Routing)
 - Einsatz von Access-Control-Lists (ACL) zur Traffic-Regulierung
- Einsatz IPv4-Multicasting in Netzen
 - Multicast Audio und Video-Streaming im Netz mit dem VLC-Mediaplayer
 - Re-Encoding und Übertragung von Streams per IP-Multicasting (IP-TV)
 - Multicast-Streaming im Batchbetrieb (einfacher TV-Betrieb)
 - Multicast-Routing in Netzen mit Cisco-Routern
 - Erarbeitung von Grundlagen der Multicast-Protokolle IGMP und PIM
 - Einrichtung eines Router- bzw. Netzwerk-Simulators ("GNS3")
 - * Basiskonfiguration eines (virtuellen) Cisco IOS-Routers
 - Einrichtung einer exemplarischen Netzstruktur auf einem Cisco-Router
 - * Ankopplung einer Virtuellen Maschine (VM) an den Simulator
 - Cisco-Router IP-Multicast Routing Konfiguration auf Basis von PIM
 - * Funktionstest mittels Live-Stream
- Wahl eines individuellen Vertiefungsthemas unter Nutzung der eigenen LAN-Infrastruktur, wie z.B.

- Einsatz von Embedded Systems im Netz (z.B. Messwerterfassung mit dem RaspberryPi)
- Multimedia-Übertragung und Streaming im Netz (mit Einsatz eines Streaming-Servers)
- Aufbau einer Controller-basierten WLAN-Infrastruktur (inkl. Gast-Portalfunktion)
- Einrichtung einer UTM-Firewall (Virtuelle Maschine, z.B. von Watchguard, ...), inkl. VPN-Einwahl
- Aufbau einer redundanten Layer-2 bzw. Layer-3 Netzinfrastruktur (per STP-Technik bzw. VRRP)
- Weitere Themen je nach Interessenlage und Aktualität.

Literatur

- TANNENBAUM, Andrew S.:
Computer Netzwerke.
5. Aufl. München: Pearson Education, 2012, ISBN 978-3-86894-137-1
- HALSALL, Fred:
Computer Networking and the Internet.
5. Aufl. München: Addison-Wesley, 2005, ISBN 978-0-32126-358-2
- HALSALL, Fred:
Multimedia Communications.
1. Aufl. München: Addison-Wesley, 2001, ISBN 0-201-39818-4
- DITTRICH, Jens; VON THIENEN, Uwe:
Netzwerk-Infrastrukturen.
Bonn: MITP-Verlag, 2002, ISBN 978-3826640919
- LAGLER, Gerhard:
Das kleine 1x1 der strukturierten Verkabelung.
Berlin: VDE-VERLAG, 2008, ISBN 978-3-7785-4046-6
- TIKART, Andreas:
Cisco Firewall - das Experimentierbuch.
1. Aufl. Bonn: MITP-Verlag, 2003, ISBN 3-8266-1374-0
- TIKART, Andreas:
Cisco Router - das Experimentierbuch.
1. Aufl. Bonn: MITP-Verlag, 2003, ISBN 3-8266-0986-7
- CISCO Router Configuration Handbook.
2nd Ed.: Cisco Press 2010, ISBN 978-1-58714-116-4
- ODOM, Wendell; WILKINS, Sean:
CCNA Routing and Switching 200-120. Official Cert Guide and Simulator Library.
Cisco Systems, 2013, ISBN 978-1-58720-466-1

- CISCO SYSTEMS: Internetworking Technology Handbook.
<http://www.cisco.com/c/en/us/td/docs/internetworking/technology/handbook/itodoc.html>
Aktualisierungsdatum 29.06.2014
- CISCO SYSTEMS:
Small Business SG-Series - Maintain and Operate Guides.
<http://www.cisco.com/c/en/us/support/switches/small-business-300-series-managed-switches/products-maintenance-guides-list.html> Aktualisierungsdatum 29.06.2014

I.1.36 Diskrete Systeme

B134 Diskrete Systeme

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B134
Bezeichnung	Diskrete Systeme
Lehrveranstaltung(en)	B134a Rechnergestützter Entwurf digitaler Systeme B134a Prakt. Rechnergestützter Entwurf digitaler Systeme B134b Diskrete Regelungstechnik
Verantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul "Diskrete Systeme" baut auf den in den Modulen "Regelungstechnik", "Einführung in Digitaltechnik", "Rechnerstrukturen und Digitaltechnik" und "Übertragungstechnik" erworbenen Kenntnissen und Fähigkeiten auf. Diese werden soweit verallgemeinert und vertieft, dass die Studierenden in die Lage versetzt werden, komplexe diskrete Systeme zu analysieren und zu entwickeln. Das Modul ist eine sinnvolle Ergänzung zu den Modulen, die regelungs- und digitaltechnische Grundlagen vermitteln. Das Modul hat einen direkten Bezug zum Studiengangsziel des Studiengangs "Bachelor Technische Informatik" Entwicklungs-kompetenzen beim Hardware-Entwurf mittels moderner Entwurfswerkzeuge zu erwerben und praktisch einzusetzen.
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Es werden Kompetenzen aus den Modulen "Einführung in Digitaltechnik", "Digitaltechnik und Rechnersysteme", "Informationstechnik" sowie "Regelungstechnik" vorausgesetzt.
Dauer	1

Lernziele

Das Ziel des Moduls ist es, bei den Studierenden eine vertiefende Sicht auf Diskrete und Digitale Systeme zu entwickeln. Nachdem in den zurückliegenden Lehrveranstaltungen einzelne Entwurfs- und Implementierungsaspekte separat betrachtet wurden, soll nun ein allgemeines Verständnis für das Zusammenspiel der Entwurfsschritte und Komponenten beim modernen

Systementwurf entstehen. Die im Modul "Systementwurf mit VHDL" bezogen auf konkrete Sprachen, Werkzeuge und Umgebungen vermittelten Kenntnisse werden nun verallgemeinert auf einer höheren Abstraktionsebene zusammengefasst und vertieft. Nach erfolgreichem Abschluss des Moduls, sind Studierende in der Lage, die hinter den modernen CAD-Systemen für Schaltungs- und Systementwurf stehenden Konzepte zu erfassen und im Entwurfsprozess erfolgreich anzuwenden sowie vergleichbare Konzepte in Projektarbeit selbstständig umzusetzen.

I.1.36.1 Rechnergestützter Entwurf digitaler Systeme

Lehrveranstaltung	Rechnergestützter Entwurf digitaler Systeme
Dozent(en)	Sergei Sawitzki
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	1.5
Prüfungsform	Mündliche Prüfung
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, Tafel

Lernziele

Die Studierenden ...

- erlangen ein tiefgründiges Verständnis der Methoden und Algorithmen des modernen Schaltungs- und Systementwurfs und Testung
- lernen die in den modernen Entwurfssystemen eingesetzten EDA-Algorithmen kennen
- erlangen die Fähigkeit, den Schaltungsentwurfsprozess von der Spezifikation bis zur technischen Umsetzung zu begreifen und in diesen bei Bedarf gezielt eingreifen zu können
- erlangen die Fähigkeit, EDA-Algorithmen zu entwerfen bzw. zu verändern

Inhalt

- Entwurfsprozess, Entwurfsfluss und Entwurfsschritte
 - Systematik und Einordnung
 - X- und Y-Diagramm
 - Schaltungstechnologien und Entwurfsstile
- Synthese und Logikoptimierung
- Technology Mapping
- Partitionierung
- Floorplanning
- Platzierung
- Verdrahtung
 - Globalverdrahtung
 - Detailverdrahtung
- Modellierung und Analyse der Zeitverhaltens
- Fehlermodellierung und Testsatzerzeugung

Literatur

- Lienig, Jens: Layoutsynthese elektronischer Schaltungen - Grundlegende Algorithmen für die Entwurfsautomatisierung, Springer Verlag 2006
- Sherwani, Naveed A.: Algorithms for VLSI Physical Design Automation, 3rd edition, Springer 1998
- Jansen, Dirk: Handbuch der Electronic Design Automation, Hanser Verlag 2001
- Alpert Charles J.; Mehta, Dinesh P.; Sapatnekar, Sachin S.: Handbook of Algorithms for Physical Design Automation, CRC Press 2009
- Hachtel Gary D.; Somenzi, Fabio: Logic Synthesis and Verification Algorithms, Springer 2006
- Betz, Vaughn; Rose, Jonathan; Marquardt, Alexander: Architecture and CAD for Deep-Submicron FPGAs, Kluwer Academic Publishers, 1999

I.1.36.2 Prakt. Rechnergestützter Entwurf digitaler Systeme

Lehrveranstaltung	Prakt. Rechnergestützter Entwurf digitaler Systeme
Dozent(en)	Sergei Sawitzki
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Praktikum
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Mündliche Prüfung
Sprache	deutsch
Lehr- und Medienform(en)	studentische Arbeit am Rechner

Lernziele

Die Studierenden vertiefen die in der Lehrveranstaltung Rechnergestützter Entwurf digitaler Systeme erworbenen Kompetenzen durch Implementierung eines Entwurfsschritts in Software und quantitative Analyse der Ergebnisse.

Inhalt

- Einführung
 - Vorstellung der Aufgabenstellung
 - Einarbeitung in die Entwurfswerkzeuge
- Umsetzung und Dokumentation der Aufgabe

Literatur

Sawitzki, Sergei: Einleitung und Aufgabenstellung zum Praktikum Rechnergestützter Entwurf digitaler Systeme, internes Dokument

I.1.36.3 Diskrete Regelungstechnik

Lehrveranstaltung	Diskrete Regelungstechnik
Dozent(en)	Carsten Burmeister
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Vorlesung mit integrierter Übung/Workshop/Assigm.
Semesterwochenstunden	2
ECTS	1.5
Prüfungsform	Portfolio-Prüfung
Sprache	deutsch
Lehr- und Medienform(en)	Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden haben ...

- eine tiefgehende Kenntnis diskreter Methoden der Regelungstechnik zum energiesparenden Einsatz in Leistungsregelungen und zum Einsatz in der Prozessautomatisierung.
- ein Verständnis für die abstrakten Beschreibungsformen komplexer diskreter Regelsysteme.
- die Fähigkeit, komplexe Systeme zu analysieren, in mehrdimensionalen Strukturen zu beschreiben und passende Regelalgorithmen für vorgegebene Aufgabenstellungen zu entwickeln.

Inhalt

- Digitale Regelung
- Abtastsysteme
- Diskrete Regelalgorithmen (Direct Digital Control)
- Fuzzy-Regelung

Literatur

- Franklin, G., Powell, J.: Feedback Control of Dynamic Systems. Pearson Education Limited 2019, 8. Auflage.

I.1.37 Projekt Eingebettete Systeme

B135 Projekt Eingebettete Systeme

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B135
Bezeichnung	Projekt Eingebettete Systeme
Lehrveranstaltung(en)	B135a Projekt Mikrocontroller B135b Prakt. PCB-Design B135c Prakt. Schaltungstechnik
Verantwortliche(r)	Dipl.-Ing. (FH) Timm Bostelmann
Zuordnung zum Curriculum	Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul ist mit den Modulen "Elektronik", "Halbleiterschaltungstechnik" und "Systemnahe Programmierung" zu kombinieren. Es passt darüber hinaus thematisch zu den Modulen "Großintegrierte Systeme" und "Echtzeitsysteme".
Semesterwochenstunden	6
ECTS	5.0
Voraussetzungen	Kenntnisse in Elektronik, Halbleiterschaltungstechnik und systemnaher Programmierung, werden vorausgesetzt.
Dauer	1

Lernziele

Nach Abschluss dieses Moduls können die Studierenden eingebettete Systeme ausgehend von einer Spezifikation planen, entwickeln, testen und prototypisch herstellen. Die hierfür nötigen systematischen Kompetenzen (wie Vorgehensweise und Einarbeitung in Schnittstellen mit Hilfe von Datenblättern), sowie die Mikrocontroller-spezifischen Kompetenzen werden im Projekt Mikrocontroller erlernt und angewendet. Die Kompetenz zur Entwicklung einfacher elektronischer Schaltungen als Schnittstelle zum Mikrocontroller, sowie zur Herstellung solcher Schaltungen als Platine werden in den Praktika PCB-Design und Schaltungstechnik erlernt.

I.1.37.1 Projekt Mikrocontroller

Lehrveranstaltung	Projekt Mikrocontroller
Dozent(en)	Timm Bostelmann
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Projekt
Semesterwochenstunden	4
ECTS	3.0
Prüfungsform	Abnahme
Sprache	deutsch/englisch
Lehr- und Medienform(en)	Handout, Softwaredemonstration, studentische Arbeit am Rechner, Tafel

Lernziele

Nachdem Studierende die Veranstaltung besucht haben, können sie ...

- sich in ein vorgegebenes oder selbst gewähltes Projektthema einarbeiten und dessen Realisierung planen.
- analoge und digitale Hardware (je nach Projektthema) als Schnittstelle eines Mikrocontrollers entwerfen.
- Software für einen Mikrocontroller entwickeln, testen und in Betrieb nehmen.
- ihre Lösung auf System- und Implementierungsebene in einer schriftlichen Ausarbeitung dokumentieren.

Inhalt

- Praktische Unterweisung an den Laborgeräten
- Planung der Projektdetails und des Projektablaufs
- Hardware-Schaltungsentwurf inklusive eines Mikrocontrollers
- Softwareerstellung für den eingesetzten Mikrocontroller
- Inbetriebnahme der Schaltung auf einem Steckbrett
- Test und empirischer Nachweis der Funktion
- Technische Dokumentation des Projektes

Literatur

abhängig vom Projektthema

I.1.37.2 Prakt. PCB-Design

Lehrveranstaltung	Prakt. PCB-Design
Dozent(en)	Dieter Opitz
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	1
ECTS	1.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden erlangen ...

- Kenntnisse und Fertigkeiten zur Verwendung der Methoden des Leiterplattenentwurfs.
- Theoretische und praktische Kenntnisse bezüglich der Bedienung eines PCB-Tools.
- Fähigkeit zum selbständigen Entwurf, Fertigung, Bestückung und Inbetriebnahme einer Leiterplatte.

Inhalt

- Leiterplattentechniken
- Einführung in ein PCB-Tool
- Durchführung eines Leiterplattenentwurfs
- Fertigung der Leiterplatte
- Inbetriebnahme der Leiterplatte

Literatur

Handbücher zum PCB-Tool

I.1.37.3 Prakt. Schaltungstechnik

Lehrveranstaltung	Prakt. Schaltungstechnik
Dozent(en)	Dieter Opitz
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Übung/Praktikum/Planspiel
Semesterwochenstunden	1
ECTS	1.0
Prüfungsform	Praktikumsbericht / Protokoll
Sprache	deutsch
Lehr- und Medienform(en)	

Lernziele

Die Studierenden erlangen die Fähigkeit ...

- zur Anwendung und Überprüfung des theoretischen Wissens.
- im Umgang mit realen Elektronikkomponenten und den Methoden der elektrischen Messtechnik.
- zur Protokollierung der Ergebnisse und zur Anfertigung einer Versuchsdokumentation.

Inhalt

Wechselnde, ausgewählte Versuche aus dem Stoffgebiet der Vorlesung Elektronik.

Literatur

Datenblätter der verwendeten Bauteile

I.1.38 Systementwurf mit VHDL

B143 Systementwurf mit VHDL

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B143
Bezeichnung	Systementwurf mit VHDL
Lehrveranstaltung(en)	B143a Systementwurf mit VHDL B143b Workshop VHDL
Verantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	Technische Informatik (Bachelor)
Verwendbarkeit	Das Modul "Systementwurf mit VHDL" baut auf den in den Modulen "Einführung in Digitaltechnik", "Rechnerstrukturen und Digitaltechnik" und "Informationstechnik" erworbenen Kenntnissen und Fähigkeiten auf und konkretisiert diese am Beispiel einer (in der Industrie als Standard geltenden) Hardwarebeschreibungssprache. Das Modul sollte mit Grundlagenmodulen kombiniert werden, die Basiskompetenzen aus den Bereichen Digitaltechnik und Aufbau von Rechnersystemen vermitteln. Das Modul hat einen direkten Bezug zum Studiengangziel des Studiengangs "Bachelor Technische Informatik" Entsicklungskompetenzen beim Hardware-Entwurf mittels moderner Entwurfswerkzeuge zu erwerben und praktisch einzusetzen.
Semesterwochenstunden	4
ECTS	5.0
Voraussetzungen	Es werden Kompetenzen aus den Modulen "Einführung in Digitaltechnik", "Digitaltechnik und Rechnerstrukturen" und "Informationstechnik" vorausgesetzt.
Dauer	1

Lernziele

Das Modul "Systementwurf mit VHDL" ist einem fortgeschrittenen Semester zugeordnet und baut somit auf Kompetenzen des zurückliegenden Studiums auf. Die Studierenden lernen den Prozess des modernen Systementwurfs von der Seite des Entwicklers kennen, indem sie eine Einführung in die Hardwarebeschreibungssprache VHDL bekommen und in dieser eigenständig eine komplexe Aufgabe lösen müssen. Der Arbeit mit VHDL dient gleichzeitig als Fallbeispiel zur Erläuterung von Entwurfsflüssen sowie Entwurfs- und Implementierungsstilen beim modernen Hardware-Systementwurf. Die Architektursicht wird durch weitere Vertiefung der

Kenntnisse von Schaltungsstrukturen ergänzt, vermittelt durch das "VHDL-Workshop". Somit lernen die Studierende das in den Modulen "Einführung in Digitaltechnik", "Digitaltechnik und Rechnerstrukturen" und "Informationstechnik" vermittelte Wissen in der Praxis einzusetzen. Nach dem Absolvieren des Moduls haben die Teilnehmer die wichtigsten Kenntnisse und Fertigkeiten erworben, die sie für einen Berufseinstieg als Hardware-Entwickler benötigen.

I.1.38.1 Systementwurf mit VHDL

Lehrveranstaltung	Systementwurf mit VHDL
Dozent(en)	Sergei Sawitzki
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Klausur
Sprache	deutsch
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, Tafel

Lernziele

Die Studierenden ...

- erlangen die Fähigkeit, ihre Kenntnisse der digitalen Grundkomponenten in einfache bis mittelkomplexe digitale Systeme unter Berücksichtigung realer Randbedingungen (Bausteinauswahl, Zeitverhalten, elektrische Anforderungen, Fertigung, Preis) umzusetzen
- erlangen die Kenntnisse über den vollständigen Entwurfsfluss zur Entwicklung digitaler Systeme unter Einsatz von Hardwarebeschreibungssprachen
- lernen die Hardwarebeschreibungssprache VHDL mit ihren unterschiedlichen Entwurfsmethoden (funktionale Beschreibung, Registertransferbeschreibung, Strukturbeschreibung) und deren Einschränkungen bezüglich der Schaltungssynthese kennen
- bekommen die Kenntnisse über Simulation, Verifikation und Validierung digitaler Systeme, über Synthese und deren Einschränkungen vermittelt
- erlangen die Kompetenz, ein durch ein Pflichtenheft spezifiziertes Problem in eine reale Hardware umzusetzen

Inhalt

- Einleitung, Systementwurf und Entwurfsfluss
- Modellaufbau
- Basiselemente
- Sprachkonstrukte
- Objekte
- Entwurfseinheiten
- Modellierungstechniken
- Simulation
- Weiterführende Konzepte (Dateien, Zeiger, Funktionen, Überladung)
- Synthese

Literatur

- Reichardt, Jürgen; Schwarz, Bernd: VHDL-Synthese: Entwurf digitaler Schaltungen und Systeme, 5. Auflage Oldenbourg Verlag 2012
- Ashenden, Peter: The Designers Guide to VHDL, 3rd edition Elsevier 2008

I.1.38.2 Workshop VHDL

Lehrveranstaltung	Workshop VHDL
Dozent(en)	Timm Bostelmann
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Workshop
Semesterwochenstunden	2
ECTS	3.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	Handout, Softwaredemonstration, studentische Arbeit am Rechner, Tafel

Lernziele

Nachdem Studierende die Veranstaltung besucht haben, können sie ...

- eine formale Anforderung in einem hierarchisch strukturierten Hardwaredesign umsetzen.
- die Richtigkeit eines solchen Designs durch Simulation (funktional und Gate-Level) nachweisen.
- ein solches Design auf einem FPGA in Betrieb nehmen.
- ihre Lösung auf System- und Implementierungsebene in einer schriftlichen Ausarbeitung dokumentieren.

Inhalt

- Einführungsaufgabe
 - Eine geführte Aufgabe bietet einen Einstieg in den Umgang mit den verwendeten Entwicklungswerkzeugen
- Aufgabe 1
Eine in Umfang und Anspruch kleine Aufgabe
- Aufgabe 2
Eine in Umfang und Anspruch größere Aufgabe
- Aufgabe 3
Zusammenführung und Ergänzung der Module aus Aufgabe 1 und Aufgabe 2 zu einer Gesamtfunktion

Literatur

- Reichardt, Jürgen; Schwarz, Bernd:
VHDL-Synthese: Entwurf digitaler Schaltungen und Systeme, 6. Auflage
Oldenbourg Verlag 2012
- Ashenden, Peter:
The Designers Guide to VHDL. 3rd edition
Elsevier 2008

- Lehmann, Gunther; Wunder, Bernhard; Selz, Manfred:
Schaltungsdesign mit VHDL
Franzis Verlag 1994

I.1.39 Laborprojekt

B133 Laborprojekt

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B133
Bezeichnung	Laborprojekt
Lehrveranstaltung(en)	B133b Laborprojekt B133a Projektmanagement
Verantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	Technische Informatik (Bachelor)
Verwendbarkeit	<p>Das Modul "Laborprojekt" beinhaltet eine individuell gestaltete Aufgabenstellung. Dadurch kann es (anhängig vom Schwerpunkt der Aufgabenstellung) auf unterschiedlichen Kompetenzen und Fähigkeiten aufbauen. Beispielhaft könnten die Kompetenzen und Fähigkeiten der Module "Elektronik", "Halbleiterschaltungstechnik", "Systemnahe Programmierung", "Algorithmen und Datenstrukturen" sowie "Echtzeitsysteme" genannt werden. Es sind aber auch weitere Kompetenzen und Fähigkeiten denkbar. Das Modul ist ein Teil der Vorbereitung auf die (in der Regel im darauf folgenden Semester zu absolvierende) Bachelor-Thesis. Das Modul hat einen direkten Bezug zum Studiengangsziel des Studiengangs "Bachelor Technische Informatik" Kompetenzen in projektorientierter Lösung von Problemen an der Schnittstelle zwischen Hard- und Software zu erwerben.</p>
Semesterwochenstunden	6
ECTS	10.0
Voraussetzungen	<p>Die Voraussetzungen sind projektbezogen abhängig vom Schwerpunkt der Aufgabenstellung und können daher nicht detailliert umschrieben werden. Im Allgemeinen werden solide Kompetenzen, die in zurückliegenden Semester erworben worden sind, erwartet, insbesondere in den Bereichen Informationstechnik, Digitaltechnik, Elektronik, Systementwurf, Algorithmen und Datenstrukturen, Echtzeitsysteme, Simulation und Regelungstechnik sowie Projektmanagement.</p>
Dauer	1

Lernziele

Ziel des Laborprojekts ist es, den Studierenden in einem fortgeschrittenen Semester einen fachlichen Rahmen für die Anwendung der bis dahin erworbenen Kenntnisse und Fertigkeiten zu liefern. Die Durchführung des Projektes kann sowohl einzeln als auch in kleineren Teams erfolgen. Durch die hohe Bewertung des Arbeitsaufwandes mit 8 ECTS bietet das Laborprojekt die Möglichkeit, auch komplexere Vorhaben umzusetzen. Idealerweise enthält das Projekt Teilaufgaben sowohl zum Software- als auch zum Hardware-Entwurf und ist an die Projekte der späteren beruflichen Praxis angelehnt. Die meisten Aufgabenstellungen dienen der Verbesserung und dem Ausbau der Laborinfrastruktur der FH Wedel (daher auch die Namensgebung). Somit werden einerseits die Studierenden der technischen Informatik in optimaler Art auf das Betriebspraktikum und die Abschlussarbeit vorbereitet sowie andererseits die Labore gepflegt und auf technisch aktuellem Niveau gehalten.

I.1.39.1 Laborprojekt

Lehrveranstaltung	Laborprojekt
Dozent(en)	Sergei Sawitzki
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Projekt
Semesterwochenstunden	4
ECTS	8.0
Prüfungsform	Abnahme
Sprache	deutsch
Lehr- und Medienform(en)	interaktive Entwicklung und Diskussion von Modellen, studentische Arbeit am Rechner

Lernziele

Ziel des Laborprojektes ist die Anwendung und Vertiefung der bisher erworbenen Kenntnisse in einem fachübergreifenden Projekt. Abhängig vom Umfang des Projektes erfolgt die Durchführung in Einzelarbeit oder in kleinen Teams. Die Bewertung mit 8 ECTS Punkten ermöglicht es den Studierenden das Laborprojekt in der Regel von der ersten Spezifikation bis zum funktionsfähigen Prototypen inklusive Erstellung der technischen Dokumentation durchzuführen. Hierdurch erhalten die Studierenden einen Gesamtüberblick über den Projektablauf und werden auf die Abschlussarbeit, sowie die berufliche Praxis vorbereitet. Das Projekt sollte der Verbesserung oder dem Erhalt der Laborinfrastruktur der FH Wedel dienen und wird themenabhängig vom jeweils zuständigen Mitarbeiter betreut.

Inhalt

Der Inhalt ist themenabhängig. Bisherige Projektthemen sind zum Beispiel:

- HV-Programmier- und Testgerät für AVR Mikrocontroller
- Funktionserweiterungen für das FPGA-Entwicklungssystem (Audioprozessor, analoge Ein- und Ausgabe)
- Steuerungsplatine für bürstenlose Motoren mit Drehzahlregelung
- Erprobung des Einsatzes von Evolutionsstrategien für die Prozessoptimierung im Prozessrechnerlabor

Literatur

abhängig von der Aufgabenstellung

I.1.39.2 Projektmanagement

Lehrveranstaltung	Projektmanagement
Dozent(en)	Gerrit Remané
Hörtermin	6
Häufigkeit	jährlich
Lehrform	Vorlesung
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Klausur
Sprache	None
Lehr- und Medienform(en)	

Lernziele

- Sie verstehen die spezifischen Charakteristika und Herausforderungen von Projekten (z.B. im Unterschied zu Prozessen)
- Sie können die wichtigsten Projektmanagement-Tools je Projektphase anwenden (Initiierung, Planung, Durchführung, Abschluss)
- Sie können wesentliche Konzepte und Methoden anwenden, um Mensch-bezogene Herausforderungen im Projektumfeld zu analysieren und zu lösen (z.B. Motivation, Feedback, Veränderung)
- Sie können geeignete Projektmanagement-Ansätze (Wasserfall vs. Agil) je nach Projekttyp auswählen

Inhalt

Im Rahmen der digitalen Transformation werden wiederkehrende Aufgaben zunehmend automatisiert. Einmalige Tätigkeiten hingegen lassen sich schwierig automatisieren und werden daher in Zukunft weiter an Bedeutung gewinnen. Diese einmaligen, temporären Aufgaben sind per Definition Projekte; nicht zuletzt aufgrund dieses Umstandes wird Projektmanagement eine der wichtigsten Fähigkeiten für eine erfolgreiche Karriere im 21. Jahrhundert.

Wie schwierig Projektmanagement in der Praxis ist, wird beispielsweise dadurch ersichtlich, dass mehr als 2 von 3 IT-Projekten ihre Ziele verfehlen. Auch wenn die Gründe hierfür im Einzelfall sehr unterschiedlich sein mögen, lassen diese sich doch in zwei breite Gruppen unterteilen. Zur ersten Gruppe zählen fehlende Projektmanagement-Kompetenzen wie Auswahl der Projektmethodik, Projektplanung oder Risikokontrolle. Zur zweiten Gruppe zählen ungenügende Soft Skills, um alle beteiligten Stakeholder zu managen, wie beispielsweise Motivation, Konfliktlösung oder Veränderungsmanagement.

Zielsetzung dieser Veranstaltung ist die Entwicklung wesentlicher Grundlagen in beiden Bereichen: Grundlegende Projektmanagementfähigkeiten (im engeren Sinne) sowie notwendige Softskills eines Projektleiters.

Kurzgliederung:

- Einführung in Projektmanagement
- Projektphasen (Initiierung, Planung, Durchführung, Abschluss)

- Soft Skills (Motivation, Veränderungsmanagement, Feedback, ...)
- Spezifische Ansätze (Wasserfall, Agil, Großprojekte, ...)

Literatur

- Verzuh: The Fast Forward MBA in Project Management, Fifth Edition, New Jersey, 2016
- Wysocki: Effective Project Management – Traditional, Agile, Extreme, Seventh Edition, Indianapolis, 2014
- PMI: A Guide to the Project Management Body of Knowledge (Pmbok Guide), Sixth Edition, Newton Square, 2017

I.1.40 Soft Skills

B118 Soft Skills

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B118
Bezeichnung	Soft Skills
Lehrveranstaltung(en)	B118a Assistenz B118b Communication Skills
Verantwortliche(r)	jeweiliger Dozent
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit	Die Inhalte dieses Moduls können gewinnbringend in Projekten, der Bachelor-Thesis und im täglichen Berufsleben genutzt werden. Das Modul sollte in allen Studiengängen verwendet werden.
Semesterwochenstunden	5
ECTS	5.0
Voraussetzungen	Fachliche Inhalte der ersten 4 Studiensemester
Dauer	1

Lernziele

Nach Abschluss des Moduls haben die Studierenden die Fähigkeit erworben, in Kooperation mit den Dozenten und Assistenten, ihr Wissen und ihre Erfahrungen aus früheren Veranstaltungen der Betriebswirtschaftslehre, Mathematik und Informatik an Studierende jüngerer Semester weiter zu geben. Mit zunehmender Dauer des Semesters verbinden die Studierenden Kenntnisse aus der Veranstaltung "Communication Skills" mit ihrer Assistenzfähigkeit.

I.1.40.1 Assistenz

Lehrveranstaltung	Assistenz
Dozent(en)	verschiedene Dozenten
Hörtermin	6
Häufigkeit	jedes Semester
Lehrform	Assistenz
Semesterwochenstunden	3
ECTS	3.0
Prüfungsform	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Sprache	deutsch
Lehr- und Medienform(en)	Tutorien

Lernziele

Die Studierenden entwickeln unter Anleitung eines Hochschullehrers die Fähigkeiten ...

- fachspezifische Aufgabenstellungen zu analysieren
- problemspezifische Lösungen zu konzipieren und
- als Ergebnis begründet zu präsentieren.

Inhalt

Im Rahmen der Assistenz werden die Studierenden von den Hochschullehrern mit konkreten (Teil)-Projekten betraut. Diese können ein weites Spektrum umfassen. So sind z.B. die Durchführung kleinerer empirischer Umfragen oder auch die eigenständige Recherche und Ausarbeitung spezieller Fachinhalte denkbar. Ebenso in Betracht kommen die Durchführung von Tutorien oder Übungen. Die Assistenz ist selbständig zu bearbeiten und kann die Abstimmung mit anderen Studierenden erfordern.

Literatur

keine

I.1.40.2 Communication Skills

Lehrveranstaltung	Communication Skills
Dozent(en)	Anna-Magdalena Kölzer
Hörtermin	6
Häufigkeit	jedes Semester
Lehrform	Workshop
Semesterwochenstunden	2
ECTS	2.0
Prüfungsform	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Sprache	deutsch
Lehr- und Medienform(en)	studentische Arbeit am Rechner

Lernziele

Die Studierenden verfügen nach dem Besuch der Lehrveranstaltung über folgende Kompetenzen:

- Besitz verbesserter persönlicher Soft Skills, wie sie für Studium oder Beruf erforderlich sind
- Sensibilität für menschliche Interaktionen und Betriebsprozesse
- Besitz erweiterter rhetorischer Fähigkeiten im Rahmen von Präsentationen, Vorträgen und Referaten sowie sozialer Kompetenz
- Kenntnis der Bedeutung von verbalen und nonverbalen Signalen für die eigene Kommunikation sowie die Fähigkeit, diese zu erkennen
- Fähigkeit zum angemessenen Verhalten bei Teamarbeit oder Projekten
- Fähigkeit zur Selbstdarstellung bei Bewerbungen, Interviews, Assessment-Centern.

Inhalt

Im Rahmen des Workshops werden die folgenden Inhalte behandelt:

- Selbstanalyse (Fragebogen) als Grundlage für Bewerbungen
- Rhetorik & Präsentation (Theorie und Praxis)
- Struktur und Aufbau von Bewerbungsunterlagen
- Bewerbungsprozess
- Interview (Theorie und Praxis)
- Assessment Center (Theorie)
- Persönlichkeitsfragebogen und Testverfahren (Intelligenz und Konzentration) (Praxis)
- Gruppenübungen (Praxis)

Literatur

- ARNOLD, Frank:
Management von den besten lernen.
München: Hans Hauser Verlag, 2010
- APPELMANN, Björn:
Führen mit emotionaler Intelligenz.
Bielefeld: Bertelsmann Verlag, 2009
- BIERKENBIEHL, Vera F.:
Rhetorik, Redetraining für jeden Anlass. Besser reden, verhandeln, diskutieren.
12. Aufl. München: Ariston Verlag, 2010
- BOLLES, Nelson:
Durchstarten zum Traumjob. Das ultimative Handbuch für Ein-, Um- und Aufsteiger.
2. Aufl. Frankfurt/New York: Campus Verlag, 2009
- DUDENREDAKTION mit HUTH, Siegfried A.:
Reden halten - leicht gemacht. Ein Ratgeber.
Mannheim/Leipzig: Dudenverlag, 2007
- GRÜNING; Carolin; MIELKE; Gregor:
Präsentieren und Überzeugen. Das Kienbaum Trainingskonzept.
Freiburg: Haufe-Lexware Verlag, 2004
- HERTEL, Anita von:
Professionelle Konfliktlösung. Führen mit Mediationskompetenz.
Handelsblatt, Bd., 6, Kompetent managen.
Frankfurt: Campus Verlag, 2009
- HESSE, Jürgen; SCHRADER, Hans Christian:
Assessment-Center für Hochschulabsolventen.
5. Auflage, Eichborn: Eichborn Verlag, 2009
- MENTZEL, Wolfgang; GROTZFELD, Svenja; HAUB, Christine:
Mitarbeitergespräche.
Freiburg: Haufe-Lexware Verlag, 2009
- MORITZ, Andr; RIMBACH, Felix:
Soft Skills für Young Professional. Alles was Sie für ihre Karriere wissen müssen.
2. Aufl. Offenbach: Gabal Verlag, 2008
- PERTL, Klaus N.:
Karrierefaktor Selbstmanagement. So erreichen Sie ihre Ziele.
Freiburg: Haufe-Verlag, 2005
- PORTNER, Jutta:
Besser verhandeln. Das Trainingsbuch.
Offenbach: Gabal Verlag, 2010
- PÜTTJER, Christian; SCHNIERDA, Uwe:
Assessment-Center. Training für Führungskräfte.
Frankfurt/New York: Campus Verlag, 2009

- PÜTTJER, Christian; SCHNIERDA, Uwe:
Das große Bewerbungshandbuch.
Frankfurt: Campus Verlag, 2010
- SCHULZ VON THUN, Friedemann; RUPPEL, Johannes; STRATMANN, Roswitha:
Miteinander Reden. Kommunikationspsychologie für Führungskräfte.
10. Auflage, Reinbek bei Hamburg: rororo, 2003

I.1.41 Auslandssemester

B099 Auslandssemester

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B099
Bezeichnung	Auslandssemester
Lehrveranstaltung(en)	B099a Auslandssemester
Verantwortliche(r)	Dipl.-Soz. (FH) Nicole Haß
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Studierende sammeln sprachliche Erfahrungen und erweitern ihre sozialen Kompetenzen, die sie in ihr Berufsleben nach Studiumabschluss einbringen können.
Semesterwochenstunden	25
ECTS	30.0
Voraussetzungen	Für eine Zulassung müssen alle Übergangsleistungen gemäß § 16a der Prüfungsverfahrensordnung und insgesamt mindestens 45 ECTS-Punkte erfolgreich absolviert sein.
Dauer	1

Lernziele

Die fachlichen Lernziele dieses Moduls werden von den ausländischen Hochschulen festgelegt. Die FH Wedel prüft, ob diese Ziele inhaltlich vergleichbar und aner kennbar sind mit den eigenen Zielen.

Im Bereich soziale Kompetenz ist das Ziel das Kennenlernen einer anderen sprachlichen und kulturellen Umgebung und das Arbeiten und Kommunizieren in dieser. Außerdem natürlich das Erlernen und/oder Festigen einer Fremdsprache.

I.1.41.1 Auslandssemester

Lehrveranstaltung	Auslandssemester
Dozent(en)	Nicole Haß
Hörtermin	6
Häufigkeit	jedes Semester
Lehrform	Veranstaltungen an ausländischer Hochschule
Semesterwochenstunden	25
ECTS	30.0
Prüfungsform	Ausland
Sprache	deutsch
Lehr- und Medienform(en)	Keine

Lernziele

Nach Abschluss des Auslandssemester besitzen die Studierenden ...

- fundierte Sprachkompetenzen in englischer, französischer oder spanischer Sprache.
- erweiterte Kenntnisse über die Kultur des Gastlandes.

Inhalt

Verpflichtendes Auslandssemester:

Für ein verpflichtendes Auslandssemester muss der Umfang der erfolgreich zu erbringenden Leistungen (ohne Englisch-Sprachkurs) mindestens 30 ECTS-Punkte betragen oder einen entsprechenden gleichwertigen Umfang in lokalen Credits aufweisen. An der ausländischen Hochschule sind fachspezifische weiterführende und keine Grundlagenkurse zu belegen. Diese sollen im Zusammenhang mit dem Wedeler Studiengang stehen (hinsichtlich der zu belegenden Fächer gemäß Modulhandbuch).

Freiwilliges Auslandssemester:

Für ein freiwilliges Auslandssemester ist der Umfang der zu leistenden ECTS-Punkte (bzw. der gleichwertige Umfang in lokalen Credits) in der jeweiligen Studienordnung vorgegeben. An der ausländischen Hochschule sind fachspezifische Kurse zu belegen, die mit dem in Wedel belegten Studiengang in ergänzendem Zusammenhang stehen. Das Studienprogramm wird vor der Abreise individuell mit dem International Office vereinbart.

Literatur

Abhängig von der ausländischen Hochschule

I.1.42 Praxissemester (dual)

B176 Praxissemester (dual)

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B176
Bezeichnung	Praxissemester (dual)
Lehrveranstaltung(en)	B176a Praxissemester (dual)
Verantwortliche(r)	Dipl.-Kauff. (FH) Journalistin Anke Amsel
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Verwendung der erworbenen Fähigkeiten in der späteren praxisorientierten Bachelor-Thesis.
Semesterwochenstunden	20
ECTS	25.0
Voraussetzungen	<p>Voraussetzung für die Zulassung zum Praxissemester ist der Nachweis der vorherigen studienbegleitenden Praxisblöcke. Sie dienen der Einarbeitung in die betriebliche Praxis. Die Praxisblöcke sind in Form von Berichten zu dokumentieren.</p> <p>Für eine Zulassung müssen alle Übergangsleistungen gemäß Studienordnung und insgesamt mindestens 75 ECTS-Punkte erfolgreich absolviert sein. Das Praxissemester darf nicht vor dem lt. Studienverlaufsplan festgelegten Semester angetreten werden. Es kann auf Antrag an den Prüfungsausschuss vorgezogen werden, wenn zu erwarten ist, dass die beziehungsweise der Studierende zum Zeitpunkt der Aufnahme des Praxissemesters die gemäß Studienordnung bis zum Praxissemester zu erwerbenden ECTS-Punkte erworben haben wird.</p>

Die Anmeldung des Praxissemesters erfolgt bei der Koordinierungsstelle „Duale Studiengänge“ über ein Formblatt.

Dauer

1

Lernziele

Das Praxissemester bietet den Studierenden die Möglichkeit eine Verbindung von studien-gangsspezifischem und unternehmensspezifischem Kompetenzprofil herauszubilden.

Die Studierenden sollen im Kooperationsunternehmen in einer Vielzahl von Tätigkeitsfeldern qualifiziert an einem größeren Projekt mit Bezug zum Studiumsziel in eigener Verantwortung unter Anleitung erfahrener Mitarbeiter mitwirken. Die projektbezogene betriebliche Tätigkeit kann sich auf mehrere unabhängige Teilprojekte erstrecken.

Dabei sollen sich die Studierenden mit Leitfragen ihres Studiengangs auseinandersetzen. Die inhaltliche Vertiefung kann durch die Einbindung des Kooperationsunternehmens teilweise über das Lehrangebot der FH Wedel hinausgehen.

Durch das projektbezogene Arbeiten werden analytische, organisatorische, kommunikative und repräsentative Techniken eingeübt sowie die Verbindung zu Anwendungsgebieten des Studien-gangs hergestellt. Ziel ist der Theorietransfer in die jeweiligen betrieblichen Funktionsbereichen. Berufliche Realität soll erlebt und erlernt werden. Die Studierenden wählen wissenschaftliche Methoden, um Aufgaben des Berufslebens zu lösen.

Den Nachweis, dass sie ihr erlerntes Wissen auf eine anwendungsbezogene Aufgabenstellung aus einem Fachgebiet selbstständig auf wissenschaftlicher Grundlage im Rahmen des festgelegten Themas anwenden können, erbringen die Studierenden im Rahmen einer wissenschaftlichen Arbeit.

I.1.42.1 Praxissemester (dual)

Lehrveranstaltung	Praxissemester (dual)
Dozent(en)	Anke Amsel
Hörtermin	6
Häufigkeit	jedes Semester
Lehrform	Praktikum
Semesterwochenstunden	20
ECTS	25.0
Prüfungsform	Praktikumsbericht / Protokoll
Sprache	deutsch
Lehr- und Medienform(en)	Keine

Lernziele

Die Studierenden ...

- können das Wissen aus dem bisherigen Studium in der Praxis anwenden und hinsichtlich der Tauglichkeit kritisch bewerten
- erweitern ihre wissenschaftlichen Ausbildung durch systematische praktische Erfahrungen
- können Projekten vorbereiten, analysieren und im Nachgang evaluieren
- bewerten Problemstellungen und können Lösungsansätze dafür entwickeln
- können Projektmanagement betreiben, Aktivitäten koordinieren, Planabweichungen hinterfragen.
- sehen und bewerten unternehmensweite und gesellschaftliche Zusammenhänge der eigenen Tätigkeit und zeigen ihre professionelle persönliche Qualifikation in der Zusammenarbeit mit Vorgesetzten und Kollegen
- nehmen Stellung zu den sozial-, datenschutz- oder umweltschutzbedingten Restriktionen bei der Umsetzung von betrieblichen Anforderungen
- reflektieren ihre Qualifikation und ihre eigene Tätigkeit
- sind zum selbständigen und eigenverantwortlichen Arbeiten fähig
- übernehmen Verantwortung für die Qualität der eigenständig übernommenen Arbeit
- entscheiden sich für systematische Vorgehensweisen und Arbeitstechniken
- klassifizieren ihre Tätigkeit zu Anwendungsgebieten des Studiengangs
- sind in der Lage das Wissen aus dem bisherigen Studium in der Praxis anzuwenden, fortzubilden und hinsichtlich der Tauglichkeit kritisch zu bewerten.

Inhalt

Der Inhalt des "Praxissemesters" muss mit der Zielrichtung des Studienganges vereinbar sein. Zur Erreichung dieses Ziel werden je Studiengang entsprechende Leitfragen formuliert. Von diesen sollen während des Praxissemester mindestens vier Themenkomplexen abgearbeitet und mindestens ein Themenkomplex vertieft werden.

Ausgehend von den Modulzielen des jeweiligen Studiengangs legt die/der Hochschulbetreuer in Absprache mit der/dem Studierenden fest, welche Themenkomplexe im Unternehmen bearbeitet werden sollen.

Literatur

themenabhängig

I.1.43 Wissenschaftliche Ausarbeitung (dual)

B179 Wissenschaftliche Ausarbeitung (dual)

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B179
Bezeichnung	Wissenschaftliche Ausarbeitung (dual)
Lehrveranstaltung(en)	B179a Wissenschaftliche Ausarbeitung (dual)
Verantwortliche(r)	Dipl.-Kauff. (FH) Journalistin Anke Amsel
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Die wissenschaftliche Ausarbeitung dient als Vorbereitung auf den wissenschaftlichen Teil der Bachelor-Thesis.
Semesterwochenstunden	3
ECTS	5.0
Voraussetzungen	<p>Voraussetzung für die Zulassung im Rahmen des Praxissemesters ist der Nachweis der vorherigen studienbegleitenden Praxisblöcke. Sie dienen der Einarbeitung in die betriebliche Praxis. Die Praxisblöcke sind in Form von Berichten zu dokumentieren.</p> <p>Für eine Zulassung müssen alle Übergangsleistungen gemäß Studienordnung und insgesamt mindestens 75 ECTS-Punkte erfolgreich absolviert sein. Das Praxissemester darf nicht vor dem lt. Studienverlaufsplan festgelegten Semester angetreten werden. Es kann auf Antrag an den Prüfungsausschuss vorgezogen werden, wenn zu erwarten ist, dass die beziehungsweise der Studierende zum Zeitpunkt der Aufnahme des Praxissemesters die gemäß Studienordnung</p>

bis zum Praxissemester zu erwerbenden ECTS-Punkte erworben haben wird.

Die Anmeldung erfolgt über ein Formblatt bei der beziehungsweise dem hochschulseitigen Betreuer(in).

Dauer

1

Lernziele

Die wissenschaftliche Ausarbeitung soll im Dualen Studium bestehende fachliche Lücken im Vergleich zum Vollzeitstudium ausgleichen und ggf. Bezüge zur betrieblichen Praxis aufweisen.

Es werden Kompetenzen zu Zitieren, Recherche, Verfassen wissenschaftlicher Texte und Methoden wissenschaftlichen Arbeitens erworben.

I.1.43.1 Wissenschaftliche Ausarbeitung (dual)

Lehrveranstaltung	Wissenschaftliche Ausarbeitung (dual)
Dozent(en)	jeweiliger Dozent
Hörtermin	6
Häufigkeit	jedes Semester
Lehrform	Praktikum
Semesterwochenstunden	3
ECTS	5.0
Prüfungsform	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Sprache	deutsch
Lehr- und Medienform(en)	Keine

Lernziele

Die Studierenden sind in der Lage, innerhalb einer vorgegebenen Frist ein Themenkomplex des Praxissemesters selbstständig wissenschaftlich zu bearbeiten und diesen kritisch zur praktischen Anwendung zu betrachten.

Inhalt

Die wissenschaftliche Arbeit ist als abschließende, vom Studierenden eigenständig aber hochschul- und unternehmensseitig betreute Ausarbeitung zum Praxissemester zu verstehen. Der Themenkomplex wird im Sinne der Zielsetzung des Praxissemesters mit der/dem hochschulseitigen Betreuer(in) abgestimmt und soll Bezüge zur betrieblichen Praxis aufweisen.

Literatur

themenabhängig

I.1.44 Betriebspraktikum

B159 Betriebspraktikum

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B159
Bezeichnung	Betriebspraktikum
Lehrveranstaltung(en)	B159a Betriebspraktikum
Verantwortliche(r)	jeweiliger Dozent
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Studierende erweitern ihre sozialen Kompetenzen und ihre Kontakte zu Unternehmen. Beides können sie nach ihrem Studiumsabschluss gewinnbringend für eine Bewerbung oder das Einleben bei ihrem späteren Arbeitgeber verwenden.
Semesterwochenstunden	0
ECTS	17.0
Voraussetzungen	Keine
Dauer	1

Lernziele

Die Studierenden sammeln Erfahrungen für die spätere berufliche Tätigkeit in einem Unternehmen.

Dies dient der Stärkung der beruflichen und sozialen Kompetenzen: Selbständigkeit, eigenverantwortliches Handeln, Kommunikation, Teamfähigkeit und Zeitmanagement.

I.1.44.1 Betriebspraktikum

Lehrveranstaltung	Betriebspraktikum
Dozent(en)	jeweiliger Dozent
Hörtermin	7
Häufigkeit	jedes Semester
Lehrform	Betriebliches Praktikum
Semesterwochenstunden	0
ECTS	17.0
Prüfungsform	Praktikumsbericht / Protokoll
Sprache	deutsch
Lehr- und Medienform(en)	

Lernziele

Das Betriebspraktikum ist ein zentraler Baustein für die berufliche Profilbildung der Studierenden. Es ermöglicht im Rahmen des Studiums einen direkten Kontakt zu Unternehmen, die von den Studierenden eigenständig angesprochen werden. Der Kontakt zum Unternehmen soll helfen, die im bisherigen Verlauf des Studiums angeeignete Fach- und Methodenkompetenz auf ausgewählte Abläufe und Problemstellungen des betrieblichen Alltags zu übertragen. Hierbei werden auch soziale Kompetenzen erprobt und gefestigt.

Inhalt

Das Betriebspraktikum soll vertieften Einblick in Prozesse und Aufbau eines Betriebes geben. Der oder die Studierende sucht sich das Betriebspraktikum mit Hilfe der Praktikadatenbank der Fachhochschule Wedel oder anderen Informationsquellen (z.B. Aushänge, Internetseiten des Wedeler Hochschulbundes). Bei Problemen bietet die Hochschule Hilfestellung. Die Tätigkeit kann im Rahmen des Tagesgeschäftes oder in einer Projektarbeit durchgeführt werden. Es wird aus Sicht der Hochschule angestrebt, dass das Betriebspraktikum als Vorlaufphase für eine sich unmittelbar anschließende Bachelorarbeit beim gleichen Unternehmen genutzt wird. Das Betriebspraktikum soll daher inhaltlich eine Brücke zur nachfolgenden Bachelorarbeit sein. Einsatzfelder sind in Absprache mit dem Unternehmen und dem oder der Dozent/in so zu wählen, dass sie auch gut geeignet sind, eine Fragestellung für eine mögliche nachfolgende Bachelorarbeit zu entwickeln.

Literatur

themenabhängig

I.1.45 Bachelor-Thesis

B150 Bachelor-Thesis

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B150
Bezeichnung	Bachelor-Thesis
Lehrveranstaltung(en)	B150a Bachelor-Thesis
Verantwortliche(r)	jeweiliger Dozent
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Keine.
Semesterwochenstunden	0
ECTS	12.0
Voraussetzungen	Voraussetzung ist das Wissen aus den Veranstaltungen der sechs vorangegangenen Semester, insbesondere der Veranstaltungen, die mit dem Themengebiet der Abschlussarbeit zusammenhängen.
Dauer	1

Lernziele

An das Betriebspraktikum schließt sich die Bachelor-Arbeit an, die sehr praxisorientiert fast ausschließlich in Unternehmen angefertigt wird und deren Themenstellung sich in enger Kooperation zwischen FH Wedel und dem jeweiligen Unternehmen in der Regel aus dem betrieblichen Umfeld ergibt.

Die Studierenden sollen mit ihrer Arbeit den Nachweis erbringen, dass sie ihr erlerntes Wissen auf eine anwendungsbezogene Aufgabenstellung aus einem Fachgebiet selbstständig auf wissenschaftlicher Grundlage im Rahmen des festgelegten Themas anzuwenden. Wesentlich sind

strukturierte und argumentierte Inhalte sowie das Einhalten üblicher Formalia.

Dies dient der Vertiefung und des konkreten Einsatzes der fachliche Kompetenzen: Methodisches Arbeiten und praktisches Anwenden der im Studium erlernten Kenntnisse.

I.1.45.1 Bachelor-Thesis

Lehrveranstaltung	Bachelor-Thesis
Dozent(en)	jeweiliger Dozent
Hörtermin	7
Häufigkeit	jedes Semester
Lehrform	Thesis
Semesterwochenstunden	0
ECTS	12.0
Prüfungsform	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Sprache	deutsch
Lehr- und Medienform(en)	

Lernziele

Die Studierenden ...

- besitzen die Fähigkeit zur Durchführung einer praxisorientierten Arbeit.
- können eine Fragestellung selbständig erarbeiten.
- können die zu erarbeitende Problematik klar strukturieren.
- können die Vorgehensweise und Ergebnisse in einer Ausarbeitung übersichtlich darstellen.
- stärken ihre praktischen Fähigkeiten im Projektmanagement-Bereich und zur Selbstorganisation.

Inhalt

Die Bachelor-Thesis soll im Regelfall in Kooperation mit einem Unternehmen erarbeitet werden. Themen aus den Arbeitsgruppen und Laboren der Hochschule sind ebenfalls möglich. Die Arbeit ist als abschließende, vom Studierenden eigenständig aber hochschul- und unternehmensseitig betreutes Projekt zu verstehen. Im Sinne der Zielsetzung der Bachelor-Ausbildung, der Erlangung des ersten berufsqualifizierenden Abschlusses, ist die Arbeit thematisch an einer Problemstellung eines kooperierenden Unternehmens orientiert oder sie besteht aus einer praxisrelevanten hochschulinternen Aufgabe.

Literatur

themenabhängig

I.1.46 Bachelor-Kolloquium

B160 Bachelor-Kolloquium

Studiengang	Bachelor-Studiengang Technische Informatik
Kürzel	B160
Bezeichnung	Bachelor-Kolloquium
Lehrveranstaltung(en)	B160a Kolloquium
Verantwortliche(r)	jeweiliger Dozent
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) Data Science & Artificial Intelligence (Bachelor) E-Commerce (Bachelor) IT-Ingenieurwesen (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit	Keine.
Semesterwochenstunden	1
ECTS	1.0
Voraussetzungen	Zulassungsvoraussetzung zum Kolloquium ist eine mit mindestens "ausreichend" bewertete Bachelor-Thesis.
Dauer	1

Lernziele

Das Kolloquium ist eine fächerübergreifende mündliche Prüfung, ausgehend vom Themenkreis der Bachelor-Thesis, und ist die letzte Prüfungsleistung, welche das Studium abschließt.

In der mündlichen Abschlussprüfung halten die Studierenden einen Fachvortrag über das von ihnen bearbeitete Thema und verteidigen ihre Bachelor-Thesis in einer anschließenden Diskussion. Dies stärkt die Fähigkeit, ein intensiv bearbeitetes Themengebiet, zusammenfassend darzustellen und professionell zu vertreten.

I.1.46.1 Kolloquium

Lehrveranstaltung	Kolloquium
Dozent(en)	jeweiliger Dozent
Hörtermin	7
Häufigkeit	jedes Semester
Lehrform	Kolloquium
Semesterwochenstunden	1
ECTS	1.0
Prüfungsform	Kolloquium
Sprache	deutsch
Lehr- und Medienform(en)	

Lernziele

Die Studierenden ...

- besitzen die Fähigkeit der konzentrierten Darstellung eines intensiv bearbeiteten Fachthemas.
- verfestigen die Kompetenz, eine fachliche Diskussion über eine Problemlösung und deren Qualität zu führen.
- verfügen über ausgeprägte Kommunikations- und Präsentationsfähigkeiten.

Inhalt

- nach Thema der Bachelor-Arbeit unterschiedlich
- Fachvortrag über das Ergebnis der Bachelor-Arbeit
- Diskussion der Qualität der gewählten Lösung
- Fragen und Diskussion zum Thema der Bachelor-Arbeit und verwandten Gebieten

Literatur

themenabhängig

Dokumenttyp	Modulhandbuch
Abschlusstyp	Bachelor
Studiengangname	Technische Informatik
Ordnungsnummer	20.0
Setzdatum	16. Dezember 2021
git	ja
git-commit	99b702a1 (lokale Änderungen vorhanden)