Vom Digital Native zum
Digital Expert

Modulhandbuch Digital Engineering

Bachelor (B.Sc.)

Wintersemester 2026/27 (SPO 2026)

Hochschule
München
University of
Applied Sciences

MUC.DAI
Munich Center for
Digital Sciences and AI

HM

06. Juli 2026

Inhaltsverzeichnis

Aufbau des Studiengangs 4

Glossar 4

101 - Computational Thinking 6

102 - Physik 9

111 - Mathematik I 11

112 - Elektrotechnik 13

114 - Technische Mechanik - Statik 15

201 - Software Engineering 17

202 - Softwareentwicklung 19

203 - Computer Systems Fundamentals 21

211 - Mathematik II 23

214 - Technische Mechanik - Kinetik 25

215 - Werkstoffkunde 26

301 - Statistik und Stochastik 28

302 - IT-Sicherheit und technischer Datenschutz 30

311 - CAD/Konstruktion 32

313 - Cyber Physical Systems 34

314 - Datenhaltung 37

315 - Fertigungstechnik 39

402 - Künstliche Intelligenz 41

411 - Numerik 43

413 - Smart Systems 45

414 - Messtechnik/Sensoren 48

415 - Thermodynamik & Fluidmechanik 50

511 - Betriebswirtschaftslehre und Projektmanagement 52

513 - Ingenieurpraktikum mit Praxisseminar 54

601 - Cloud Computing 56

611 - Visual Computing **58**

612 - Digitale Signalverarbeitung **60**

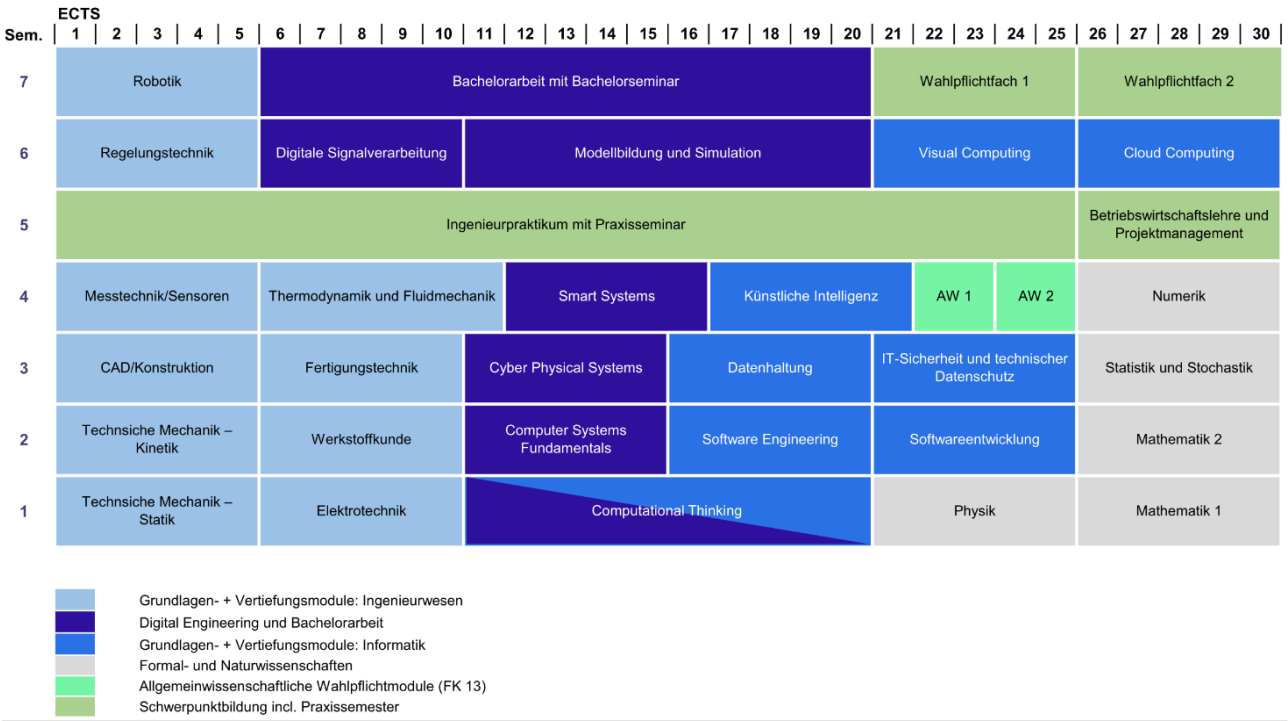
613 - Modellbildung und Simulation **62**

614 - Regelungstechnik **64**

711 - Bachelorarbeit mit Bachelorseminar **66**

712 - Robotik **68**

Aufbau des Studiengangs



Die Dauer jedes Moduls ist ein Semester.

Glossar

Lehrveranstaltungsarten (siehe ASPO § 9)

- SU: seminaristischer Unterricht
- Ü: Übung
- S: Seminar
- Pra: Praktikum

Prüfungsformen

- schrP: schriftliche Prüfung
- mdlP: mündliche Prüfung
- Präs: Präsentation
- Ref: Referat
- ModA: Modularbeit
- praP: praktische Prüfung
- FrwL: Freiwillige Prüfungsleistungen zur Verbesserung der Modulendnote
- BA: Bachelorarbeit

Weitere Abkürzungen

- SWS: Semesterwochenstunden

- FK03: Fakultät für Maschinenbau, Fahrzeugtechnik, Flugzeugtechnik
- FK04: Fakultät für Elektrotechnik und Informationstechnik
- FK06: Fakultät für Angewandte Naturwissenschaften und Mechatronik
- FK07: Fakultät für Informatik und Mathematik
- FK08: Fakultät für Geoinformation
- FK12: Fakultät für Design
- FK13: Fakultät für Studium Generale und Interdisziplinäre Studien
- FK14: Fakultät für Tourismus

101 - Computational Thinking

Empfohlenes Studiensemester: 1
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 10
Prüfungsform: ModA oder praP oder schrP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (4SWS), Pra (4SWS)
Gesamter Workload: 300 Stunden
Präsenzzeit: 60 Stunden Seminaristischer Unterricht, 60 Stunden Praktikum
Selbststudium: 180 Stunden
Modulverantwortung: Prof. Dr. Martin Hobelsberger (FK07)
Weitere Lehrende: Prof. Dr. Thomas Kofler (FK07), Prof. Dr.-Ing. Martin Orehek (FK07), Prof. Dr. Matthias Bauer (FK07), Prof. Dr. Benedikt Dietrich (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Die Studierenden erwerben die Fähigkeit, algorithmische Lösungskonzepte für Problemstellungen zu entwickeln und diese mit einer höheren Programmiersprache umzusetzen. Sie können eine Problemstellung analysieren, die zur Lösung erforderlichen Schritte identifizieren, diese Schritte verbalisieren, geeignete Konstrukte auswählen sowie syntaktisch und semantisch korrekt umsetzen. Im konkreten werden folgende Ziele adressiert:

Studierende im Kurs Computational Thinking

- entwickeln ein breites und solides Verständnis von Informatik, der Informationsverarbeitung und Programmierung.
- kennen den Grundlegenden Aufbau und die Funktionsweise von IT Systemen.
- entwickeln die Kompetenz algorithmisch zu denken um u.a.: Alltagssituationen in Algorithmen (unmissverständliche Lösungsschritte) zu übertragen, Probleme in Teilprobleme zu zerlegen, Algorithmisch abstrakte Ähnlichkeiten zu erkennen und für sich zu nutzen (Mustererkennung), wichtige von unwichtigen Informationen zur Lösung eines Problems zu unterscheiden und ihre Lösungsschritte dadurch so allgemein wie möglich zu halten.
- entwickeln die Kompetenz effiziente Lösungskonzepte für (Programmier-) Problemstellungen zu entwickeln.
- entwickeln die Fähigkeit Lösungsschritte in unmissverständlicher Form niederzuschreiben (Design von Algorithmen).
- wenden moderne Werkzeuge zur Umsetzung von Lösungskonzepten an.
- setzen Programmierprojekte eigenständig und im Team um.

Selbstkompetenz

Studierenden im Kurs Computational Thinking

- modellieren Lösungen mit Hilfe von Konstrukten in Programmiersprachen, so dass eine algorithmische Verarbeitung möglich wird.
- entwickeln eine Neugier für noch unbekannte und ungelöste Probleme.

Sozialkompetenz

Studierenden im Kurs Computational Thinking

- erarbeiten Lösungen zu anwendungsnahen Problemen im Team.
- präsentieren und erläutern Teammitgliedern ihre Lösungen.
- werden für sozial-gesellschaftliche Zusammenhänge im Rahmen der Informationsverarbeitung sensibilisiert.

Inhalt

- Konzepte und Methoden des Computational Thinking
- Grundlegende Konzepte, Arbeitsweisen und Aufbau von IT Systemen
- Grundlegende Konzepte der prozeduralen Programmierung auf der Basis einer aktuellen, allgemein verfügbaren Programmiersprache
- Einführung und Einstieg in die objektorientierte Programmierung

Im Einzelnen werden behandelt:

- Der intrinsische, sozial-gesellschaftliche und langfristige Wert des Computational Thinkings
- Informationsverarbeitung: Informationsdarstellung und Codierung (Binärsystem und andere Formate), Informationsmanipulation (Funktionsweise von Gattern bis zum Addierer), Informationskreislauf (Prinzipien und Arbeitsweise eines IT-Systems), Gegenüberstellung von Hard- und Software
- Computational Thinking verstehen als Kunstform: Unterschied zwischen konkreter Welt, modellierter Welt und abstrahierter Welt, Unterschied zwischen Programmieren und Computational Thinking betont als kreativer Akt, Definition von Quellcode, Programm und Algorithmus, Evolution unterschiedlicher Programmiersprachen
- Grundlagen der Programmierung mit u.a.: Arithmetische Ausdrücke, Variablen und Datentypen, Kontrollstrukturen, Datenstrukturen, Arrays und Listen, Strings und Textzeichen, Funktionen, Rekursion, Klassen und Methoden, Exception Handling, Tests
- Bearbeitung/Lösung konkreter algorithmischer Probleme aus den Bereichen der jeweiligen Studiengänge (z.B. Analyse und Darstellung von Datensätzen, Erstellung und Nutzung eines neuen binären Bildformats, Bildverarbeitung, Animation einer Schwingung, Zeichnen einer Schneeflocke, Nutzung eines Web Crawlers, Animation
- Einführung in Algorithmen und Datenstrukturen: Entwicklung einer Intuition für Komplexität, Tradeoff zwischen Speicher und Rechenleistung (z.B. verlustfreie Komprimierung von Daten), Sortieralgorithmen, Konzept der Hashfunktion (Dictionaries/HashMaps), mehrdimensionale Listen, Unterschied zwischen veränderlichen (mutable) und unveränderlichen (immutable) Datenstrukturen, Entwurf eigener erster einfacher Datenstrukturen (z.B. ein Namensregister), Umgang mit Datenstrukturen aus dem Bereich des Scientific Computing (z.B. Numpy-Arrays und Pandas-Datenframes)

Lehrmethoden und Lernformen

Folien, Tafel, virtuelle Lehr-Lernräume, interaktive Notebooks, interaktives Jupyter-Buch, anwendungsorientierte Projekte, Lehr-/Lernvideos, Gastvorträge

Verwendbarkeit des Moduls

Bachelor Digital Engineering, Bachelor Informatik und Design, Bachelor Geodata Science, Hochschulzertifikat Digital

Literatur

- Paul Curzon, Peter W. McOwan. Computational Thinking: Die Welt des algorithmischen Denkens - in Spielen, Zaubertricks und Rätseln. Springer Verlag
- Klein, Bernd. Einführung in Python 3: Für Ein- und Umsteiger. Carl Hanser Verlag GmbH Co KG, 2017
- Klein, Bernd. Numerisches Python: Arbeiten mit NumPy, Matplotlib und Pandas. Carl Hanser Verlag GmbH Co KG, 2019
- VanderPla, Jake. Python Data Science Handbook, O'Reilly Media, Inc. 2016

102 - Physik

Empfohlenes Studiensemester: 1
Turnus: Wintersemester
Sprachen: Deutsch, Englisch
ECTS: 5
Prüfungsform: schrP oder praP oder ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (4SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 60 Stunden Seminaristischer Unterricht
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Andreas Gubner (FK03)
Weitere Lehrende: Prof. Dr. Christian Schwarzbauer (FK06)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Studierende lernen die Grundlagen der Physik und die Herangehensweise an naturwissenschaftliche Aufgaben, sofern nicht ohnehin Bestandteil des jeweiligen weiterführenden Fachs werden folgende Ziele adressiert:

- Studierende beschreiben und erklären das SI-Einheitensystem.
- Studierende nennen die wichtigen physikalischen Themen und Gebiete und entwickeln eine Gesamtübersicht über die Physik.
- Studierende benennen wichtige technische Anwendungen physikalischer Gesetze.
- Studierende verstehen die behandelten physikalischen Grundlagen, Naturgesetze, Prinzipien und Modelle.
- Studierende sind vertraut mit der mathematischen Quantifizierung physikalischer Sachverhalte.
- Studierende wenden die Naturgesetze, Prinzipien und Modelle auf techniknahe, anwendungsorientierte physikalische Aufgabenstellungen an.
- Studierende setzen die vermittelten Inhalte auf neue Aufgabenstellungen um.

Inhalt

- Größen und Einheiten, SI-System
- Grundlagen der Atomphysik: Bohrsches Atommodell, Atomkern und Elektronenhülle, Periodensystem der Elemente, Chemische Bindung, Isotope und Radioaktivität, Massendefekt
- Gravitation
- Stoffeigenschaften fester Körper
- Grundlagen der Fluidmechanik und der Thermodynamik
- Grundlagen des Wärmetransports (Leitung, Konvektion, Strahlung)
- Einführung in die Schwingungs- und Wellenlehre

- Schall: Definition, Messbarkeit, Lärm und Lärmschutz
- Optik: Lichteigenschaften, Farbenlehre, Teilchencharakter des Lichts, geometrische Optik (Kamera etc.)
- Magnetismus

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Flipcharts, optional Ringvorlesung

Verwendbarkeit des Moduls

Bachelor Digital Engineering, Bachelor Geodata Science

Literatur

- Vorkurs Physik fürs MINT-Studium; P. Steglich, K. Heise; Springer Spektrum; ISBN 978-3-662-62125-7 ISBN 978-3-662-62126-4 (eBook); URL https://doi.org/10.1007/978-3-662-62126-4 Zugriff vom 14.05.2021
- PHYSIK-Beispiele und Aufgaben; H. Stroppe, P. Streitenberger, E. Specht; Heribert Stroppe; 2., aktualisierte und erweiterte Auflage; Carl Hanser Verlag München 2021; Print-ISBN 978-3-446-46406-3, E-Book-ISBN 978-3-446-46800-
- Physik-Eine Einführung für Ingenieure und Naturwissenschaftler; U. Harten; 7., bearbeitete und aktualisierte Auflage; Springer Vieweg 2017;ISBN 978-3-662-49753- ISBN 978-3-662-49754-8 (eBook); URL https://doi.org/10.1007/978-3-662-49754-8 Zugriff vom 14.05.2021

111 - Mathematik I

Empfohlenes Studiensemester: 1
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP oder ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (4SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 60 Stunden Seminaristischer Unterricht
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Christian Möller (FK03)
Weitere Lehrende: Prof. Dr. Laurent Demaret (FK03), Prof. Dr. Katina Warendorf (FK03), Prof. Dr. Michael Wibmer (FK03), Prof. Dr. Wolfgang Högele (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Die Studierenden sind in der Lage,

- einfache Sachverhalte in der Sprache der Mathematik zu formulieren (Modellbildungskompetenz).
- mathematische Argumentationen kritisch zu reflektieren.
- die Probleme der eindimensionalen Analysis zu klassifizieren, geeignete Lösungsverfahren auszuwählen und sie sicher, formal korrekt und kreativ einzusetzen.
- sicher mit Termen, (Un-)Gleichungen und Funktionen umzugehen.
- die Grundbegriffe der Analysis und der linearen Algebra zu benutzen, miteinander zu verknüpfen und auf andere Bereiche anzuwenden.

Inhalt

Grundlegende Konzepte, Methoden und numerische Verfahren der eindimensionalen Analysis für die folgenden Themengebiete:

- Reihen (Folgen, Konvergenz unendlicher Reihen, Taylorpolynome und -reihen, ...)
- Logische Grundlagen und Beweisverfahren, insbesondere vollständige Induktion
- Funktionen und Modelle: Polynome (Polynominterpolation, Horner-Schema , ...), Logarithmus und Exponentialfunktion, trigonometrische Funktionen, inverse Funktionen, etc.
- Differentiation und ihre Anwendung (Differentiationsregeln, Extremwertaufgaben, Newton-Verfahren, etc.)
- Integration und ihre Anwendung (numerische Integration, etc.)
- Komplexe Zahlen

Grundlegende Konzepte, Methoden und Verfahren der linearen Algebra für die folgenden Themengebiete:

- Vektorräume
- Lineare Abbildungen und Matrizen
- Lineare Gleichungssysteme
- Eigenwerte

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, interaktive Notebooks (Jupyter, Pluto)

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Arens et al., Mathematik, Springer, 2018
- Bärwolff, Höhere Mathematik für Naturwissenschaftler und Ingenieure, Springer, 2017
- Karpfinger, Höhere Mathematik in Rezepten, Springer, 2021
- Strang, Introduction to linear algebra, Wellesley-Cambridge Press, 2016

112 - Elektrotechnik

Empfohlenes Studiensemester: 1
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: praP oder schrP oder mdlP oder ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr.-Ing. Frank Palme (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Studierende lernen die Grundlagen der Elektrotechnik insbesondere mit den Lernzielen dass sie

- die Grundbegriffe und Grundgesetze der Elektrotechnik auf den Gebieten Elektrostatik, Gleichstrom, Wechselstrom, Magnetismus sowie die zugrunde liegenden physikalischen Zusammenhänge verstehen.
- die Fähigkeit zur Berechnung elektromagnetischer Felder in Vakuum und Materie, von Gleich- und Wechselstromnetzwerken sowie von magnetischen Kreisen erlangen.
- die Kompetenz entwickeln elektrische Schaltungen unter Nutzung fundamentaler Bauelemente (Spannungs- und Stromquellen, Widerstände, Kondensatoren, Spulen) zu entwerfen, zu dimensionieren und zu simulieren.
- elektrotechnische Systeme dabei insbesondere im Bewusstsein von Nachhaltigkeitszielen einsetzen.

Inhalt

- Ladung, Elektrisches Feld, Potential, Spannung, Kapazität, Energie
- Stromstärke, Stromdichte, Ohmsches Gesetz, Kirchhoffsche Regeln, Gleichstromnetzwerke, Zweipoltheorie, Leistung, Energie, Wirkungsgrad
- Magnetisches Feld, Fluss und Flussdichte, magnetische Kräfte, magnetischer Kreis, Induktionsgesetz, Induktivität, Energie
- Komplexe Wechselstromrechnung, Zeigerdiagramme, Wechselstromwiderstände, Wechselstromnetzwerke, Wirk-, Blind- und Scheinleistung
- Entwurf, Simulation und Anwendung von elektrotechnischen Bauteilen und Schaltungen anhand praxisorientierter Beispiele und Übungen

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, digitale Lehrmedien

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Rudolf Busch: Elektrotechnik und Elektronik für Maschinenbauer und Verfahrenstechniker, Vieweg+Teubner (2015)
- Gert Hagmann: Grundlagen der Elektrotechnik, Aula-Verlag (2020)

114 - Technische Mechanik - Statik

Empfohlenes Studiensemester: 1
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (4SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 60 Stunden Seminaristischer Unterricht
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Markus Gitterle (FK03)
Weitere Lehrende: Prof. Dr. Armin Fritsch (FK03), Prof. Dr. Jörg Middendorf (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

Vorkenntnisse in Mathematik (Vektorrechnung, Infinitesimalrechnung)

Lernziele

Fach- und Methodenkompetenz

Die Studierenden sind nach erfolgreichem Bestehen des Moduls in der Lage, statische Probleme an Systemen starrer Körper selbständig zu lösen. Dazu gehört, dass die Studierenden ein reales Bauteil oder eine reale technische Struktur in Form eines mechanischen Modells idealsieren, dieses Modell durch Freischneiden und der Formulierung von Gleichgewichtsbedingungen in mathematische Gleichungen übersetzen sowie diese Gleichungen lösen können. Insbesondere sind die Studierenden in der Lage, das Schnittprinzip anzuwenden, eingeprägte Kräfte und Reaktionskräfte zu erkennen (3. NEWTONsches Axiom) sowie Gleichgewichtsbedingungen aufzustellen.

Außerdem können die Studierenden elastostatische Probleme am Beispiel von Stabsystemen selbständig lösen. Dazu gehören neben der Formulierung von Gleichgewichtsbedingungen bzw. der Berechnung von Schnittgrößen die Einbeziehung von Verformungsgleichungen (z.B. die Ermittlung von Verschiebungen), bei statisch unbestimmten Systemen die Formulierung von Kompatibilitätsbedingungen und schließlich die Berücksichtigung von Randbedingungen. Zentrales Lernziel ist, dass die Studierenden die Zusammenhänge von äußeren Belastungen eines Systems und den daraus resultierenden inneren Beanspruchungen sowie den Verformungen erläutern können. Darüber hinaus sind die Studierenden in der Lage, die Voraussetzungen, Idealisierungen sowie die Grenzen der Anwendbarkeit der elementaren Stabtheorie zu beschreiben und zu beurteilen.

Inhalt

Statik starrer Körper: Gleichgewichtsbedingungen an zentralen und allgemeinen Kräftesystemen, Schwerpunkt, Lagerreaktionen, Fachwerke, Schnittgrößen an Balken.

Elastostatik (Beanspruchungen und Verformungen elastischer Körper): Spannung, Verzerrung und Elastizitätsgesetz am Beispiel des Stabes, Kräfte und Verformungen in Stäben, statisch unbestimmt gelagerte Stäbe/Stabsysteme.

Lehrmethoden und Lernformen

Tafel, Beamer, Online-Skript

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Gross/Hauger/Schröder/Wall: "Technische Mechanik 1", Springer-Verlag.
- Gross/Hauger/Schröder/Wall: "Technische Mechanik 2", Springer-Verlag.

201 - Software Engineering

Empfohlenes Studiensemester: 2
Turnus: Sommersemester
Sprachen: Deutsch, Englisch
ECTS: 5
Prüfungsform: ModA oder praP oder schrP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Benedikt Dietrich (FK07)
Weitere Lehrende: Prof. Dr. Matthias Bauer (FK07), Prof. Dr. Gudrun Socher (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Kenntnisse in mindestens einer höheren Programmiersprache

Lernziele

Fach- und Methodenkompetenz

Studierende lernen die Anwendung der grundlegenden Konzepte des Software Engineerings. Im konkreten werden folgende Ziele adressiert:

- Studierende kennen die wesentlichen Eigenschaften von Software.
- Studierende kennen grundlegende Prinzipien des Software Engineerings.
- Studierende können den Softwareentwicklungszyklus und seine Phasen beschreiben.
- Studierende können Vorgehensmodelle und ihre Eigenschaften benennen und einordnen.
- Studierende können für alle Aktivitätstypen des Software Entwicklungsprozesses geeignete Methoden und Werkzeuge vorschlagen.
- Studierende können Definition und Entwurf von Software in geeigneter Notation formulieren.
- Studierende können Software im Team entwickeln.
- Studierende können Werkzeuge zur Softwareentwicklung und zur Unterstützung des Software Entwicklungsprozesses geeignet einsetzen.

Inhalt

- methodische Entwicklung objektorientierter Softwaresysteme
- agile Vorgehensmodelle
- Einführung der Unified Modeling Language (UML) und wesentliche Diagrammtypen der UML (Use Case Diagramm, Klassendiagramm, Objektdiagramm, Sequenzdiagramm und Aktivitätsdiagramm)
- typische Arbeitsschritte der Anforderungsermittlung an Software, der Erstellung der Softwarespezifikation und des Softwareentwurfs
- Werkzeuge im Software Engineering und DevOps
- Grundlagen von Software Architekturen
- ausgewählte Entwurfsmuster

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Gastvorträge, projektbasiertes Lernen, virtuelle Lehrräume

Verwendbarkeit des Moduls

Bachelor Digital Engineering, Bachelor Informatik und Design, Bachelor Geodata Science

Literatur

- Metzner; Software-Engineering - kompakt, Hanser, 2020
- Sommerville; Software Engineering; Pearson Studium, 2015
- Kleuker; Grundkurs Software-Engineering mit UML, Springer Vieweg, 2018
- Rupp; UML 2 glasklar, Hanser, 2012

202 - Softwareentwicklung

Empfohlenes Studiensemester: 2
Turnus: Sommersemester
Sprachen: Deutsch, Englisch
ECTS: 5
Prüfungsform: praP oder schrP oder mdlP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Benedikt Dietrich (FK07)
Weitere Lehrende: Prof. Dr. Matthias Bauer (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Die Studierenden können einfache mathematische und algorithmische Problemstellungen analysieren und die zur Lösung erforderlichen Schritte identifizieren.

Lernziele

Fach- und Methodenkompetenz

Die Studierenden

- erklären in eigenen Worten die Bedeutung der Softwareentwicklung für ihren fachlichen Kontext.
- beschreiben den Unterschied zwischen einzelnen programmiersprachlichen Konstrukten.
- begründen, welches Sprachkonstrukt in welchem Kontext zu verwenden ist, und warum.
- wägen systematisch ab, welches Konzept der Programmiersprache am besten geeignet ist, um eine bestimmte Anforderung in einem Algorithmus umzusetzen.
- identifizieren Stärken und Verbesserungspotenzial in gegebenem Quelltext.
- bewerten eine von ihnen selbst erstellte Software kritisch hinsichtlich Stärken und Schwächen, die in Bezug zu grundlegenden Qualitätsanforderungen bestehen (Lesbarkeit, Testbarkeit, Korrektheit).
- entwickeln für ein einfaches Problem aus einer gegebenen Anforderungsspezifikation heraus eine Umsetzung in Software. Diese erfüllt dabei grundlegende Qualitätsanforderungen. (Ein „einfaches Problem“ ist dabei eine Aufgabenstellung, die mit maximal ca. 500 Zeilen Quelltext zu lösen ist.)
- erstellen schematisch grundlegende Testfälle.
- nutzen ein Werkzeug, um Testfälle automatisiert auszuführen.
- setzen systematisch Werkzeuge ein, die den Grad der erreichten Testabdeckung ermitteln.
- nutzen Werkzeug zur Versionsverwaltung sowie eine moderne IDE und Build-Werkzeuge.
- gleichen beim Verwenden des Debuggers das, was der Debugger anzeigt, ab mit der eigenen mentalen Erwartung, bis beides nicht mehr zueinander passt und zeigen so Soll-/Ist-Differenzen auf.
- bearbeiten Software in kleinen Teams und formulieren dabei als Feedback-Geber ihre Kritik gemäß Feedback-Regeln. Sie halten als Feedback-Nehmer beim Empfangen von Kritik die formalen Feedback-Regeln ein.

Inhalt

Grundlegende Konzepte der Softwareentwicklung auf der Basis einer aktuellen, allgemein verfügbaren Programmiersprache. Im Einzelnen werden behandelt:

- Grundkonzepte des verwendeten Programmierparadigmas (objektorientiert, funktional oder imperativ)
- Testen
- Grundlegende Algorithmen
- Ausnahmebehandlung
- Systematische Fehlersuche und Debugging

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Selbstlernmaterial, Lehr-/Lernvideos, Lesetexte

Verwendbarkeit des Moduls

Bachelor Digital Engineering, Bachelor Geodata Science

Literatur

- Bjarne Stroustrup, Programming: Principles and Practice Using C++, 2nd edition, Addison-Wesley, 2014, ISBN 9780321992789
- Robert C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship, Pearson, 2008, ISBN 9780136083238

203 - Computer Systems Fundamentals

Empfohlenes Studiensemester: 2
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: praP oder schrP oder mdlP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Martin Hobelsberger (FK07)
Weitere Lehrende: Prof. Dr. Stefan Wallentowitz (FK07), Prof. Dr.-Ing. Martin Orehek (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Grundlegendes Verständnis von Informatik, der Informationsverarbeitung und Programmierung (z.B. erworben im Modul: Computational Thinking).

Lernziele

Fach- und Methodenkompetenz

Studierende lernen die Grundlagen des Aufbaus von Rechnersystemen und die Grundlagen der Netzwerkkommunikation.
Die Studierenden

- erklären eine digitale Schaltung und können diese einordnen.
- verstehen und erkennen die Unterschiede möglicher Varianten des Rechneraufbaus.
- entwickeln Software auf Basis einer mittels Datenblätter beschriebenen HW Architektur.
- beschreiben die verschiedenen Layer bei der Netzwerkkommunikation (ISO/OSI, TCP/IP-Schichten).
- analysieren unterschiedliche Technologien für die Netzwerkkommunikation.
- entwickeln verteilte Client-Server Anwendungen.

Überfachliche Kompetenz

- Über Gruppenarbeiten werden kommunikative Kompetenzen gefördert.

Inhalt

- Grundlagen der Digitaltechnik
- Etablierte Rechnerarchitekturen (von Neumann vs. Harvard)
- Speicherhierarchien
- Grundlagen der Instruction Set Architecture (ISA) am Beispiel unterschiedlicher Plattformen (z.B. RISC-V, ARM)
- Einführung verbreiteter SoC Komponenten und deren Ansteuerung (z.B. Memory Mapped IO, GPIO, UART)

- ISO/OSI Modell mit etablierten Technologien (z.B. HTTP, TCP, UDP, IP, Ethernet)
- Anwendung der Netzwerkkommunikation am Beispiel von verteilten Client-Server Anwendungen

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Gastvorträge

Verwendbarkeit des Moduls

Bachelor Digital Engineering, Bachelor Geodata Science, Wahlpflichtmodul Bachelor Informatik und Design

Literatur

- Hoffmann D. W. : *Grundlagen der Technischen Informatik*, Carl Hanser Verlag GmbH & Co. KG
- Nisan N., Schocken S.: *The elements of computing systems: building a modern computer from first principles*, 1. MIT Press pbk. ed. Cambridge, Mass.: MIT Press.
- Bryant R. E., O'Hallaron D. R.: *Computer systems: a programmer's perspective*, Third edition. Boston Columbus Hoboken Indianapolis New York San Francisco Cape Town: Pearson.
- Patterson D. A., Hennessy J. L.: *Computer organization and design: the hardware/software interface, RISC-V edition*, Cambridge, Massachusetts: Morgan Kaufmann Publishers.
- Kurose J. F., Ross K. W.: *Computernetzwerke: Der Top-Down Ansatz*, Pearson Studium
- Tanenbaum A. S.: *Computernetze*, Pearson Studium
- Badach A., Hoffmann E.: *Technik der IP-Netze: Internet-Kommunikation in Theorie und Einsatz*, München: Hanser
- Weitere Literatur wird im Rahmen der aktuellen Veranstaltung bekannt gegeben.

211 - Mathematik II

Empfohlenes Studiensemester: 2
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP oder ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (4SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 60 Stunden Seminaristischer Unterricht
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Christian Möller (FK03)
Weitere Lehrende: Prof. Dr. Laurent Demaret (FK03), Prof. Dr. Katina Warendorf (FK03), Prof. Dr. Michael Wibmer (FK03), Prof. Dr. Wolfgang Högele (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Mathematik I

Lernziele

Fach- und Methodenkompetenz

Die Studierenden sind in der Lage,

- einfache Sachverhalte in der Sprache der Mathematik zu formulieren (Modellbildungskompetenz).
- mathematische Argumentationen kritisch zu reflektieren.
- die Probleme der mehrdimensionalen Differentialrechnung zu klassifizieren, geeignete Lösungsverfahren auszuwählen und sie sicher, formal korrekt und kreativ einzusetzen.
- gewöhnliche Differentialgleichungen zu klassifizieren und eine geeignete Lösungsmethode auszuwählen und anzuwenden.
- die Grundbegriffe der mehrdimensionalen Differentialrechnung sowie von gewöhnlichen Differentialgleichungen miteinander zu verknüpfen und in anderen Gebieten wie Statistik, Numerik, Optimierung oder Modellbildung einzusetzen.

Inhalt

Grundlegende Konzepte, Methoden und numerische Verfahren der mehrdimensionalen Analysis für die folgenden Themengebiete:

- Partielle Ableitung, Gradient, Richtungsableitung, Tangentialebene, Jacobi und Hesse-Matrix
- Differentiationsregeln, Satz von Schwarz, Taylor-Entwicklung, Linearisierung, notwendige und hinreichende Bedingungen für Extrema und Sattelpunkte
- Vektorfelder und Kurvenintegrale
- Mehrdimensionale Integrale
- Ausgewählte Integralsätze

Grundlegende Konzepte, Methoden und numerische Verfahren der gewöhnlichen Differentialgleichungen (DGL):

- Modellierung mit DGL
- DGL 1. Ordnung: allgemeine und spezielle Lösungen von separablen und linearen DGL
- DGL 2. Ordnung: allgemeine Schwingungs-DGL
- Theorie linearer DGL-Systeme

Ergänzungen zentraler Anwendungen der linearen Algebra wie beispielsweise

- Projektionen und Least Squares
- Wichtige Faktorisierungen von Matrizen (LU-Zerlegung, QR-Zerlegung, Diagonalisierung)

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, interaktive Notebooks (Jupyter, Pluto)

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Arens et al., Mathematik, Springer, 2018
- Bärwolff, Höhere Mathematik für Naturwissenschaftler und Ingenieure, Springer, 2017
- Grüne, Junge, Gewöhnliche Differentialgleichungen, Springer 2016
- Karpfinger, Höhere Mathematik in Rezepten, Springer, 2021

214 - Technische Mechanik - Kinetik

Empfohlenes Studiensemester: 2
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (4SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 60 Stunden Seminaristischer Unterricht
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Markus Gitterle (FK03)
Weitere Lehrende: Prof. Dr. Jörg Middendorf (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

Technische Mechanik - Statik

Lernziele

Fach- und Methodenkompetenz

Zentrales Lernziel ist, dass die Studierenden den Zusammenhang zwischen Kräften und Bewegungen an Systemen starrer Körper erläutern können. Die Studierenden sind nach erfolgreichem Bestehen des Moduls in der Lage, kinetische Probleme an Systemen starrer Körper selbständig zu lösen. Dazu gehören einerseits das Freischneiden der einzelnen starren Körper, die Formulierung von Schwerpunktsatz und Drallsatz, das Erkennen kinematischer Zusammenhänge bei gekoppelten Bewegungen sowie die Zeitintegration der Bewegungsgleichungen. Andererseits können die Studierenden alternativ die Bilanzierung mit Hilfe von Arbeits- und Energiesatz anwenden.

Inhalt

Kinematik des Massepunktes sowie des starren Körpers. Kinetik des Massenpunktes sowie des starren Körpers. Der Anwendungsfall bleibt auf die Ebene beschränkt. Im Einzelnen: Schwerpunktsatz, Drallsatz, Massenträgheitsmomente, Arbeitssatz und Energiesatz, Impulssatz, Haftung und Reibung.

Lehrmethoden und Lernformen

Tafel, Beamer, Online-Skript

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Gross/Hauger/Schröder/Wall: "Technische Mechanik 3", Springer-Verlag.

215 - Werkstoffkunde

Empfohlenes Studiensemester: 2
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (3SWS), Pra (1SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 45 Stunden Seminaristischer Unterricht, 15 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Matthias Reihle (FK03)
Weitere Lehrende: Prof. Dr. Henning Stoll (FK03), Prof. Dr. Ulrich Dahn (FK03), Prof. Dr. Tobias Hornfeck (FK03), Prof. Dr. Robert Ramakrishnan (FK03), Prof. Dr. Swen Zaremba (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Nach der Teilnahme am Modul sind die Studierenden in der Lage, Werkstoffstrukturen und Gebrauchseigenschaften in Berechnung, Konstruktion, Fertigung und betrieblicher Anwendung zu verknüpfen. Die Studierenden charakterisieren Werkstoffeigenschaften fachgerecht und wählen Werkstoffe entsprechend der gestellten Anforderungen und der gezielten Beeinflussung der Werkstoffeigenschaften aus.

Inhalt

- Aufbau und Struktur von metallischen Werkstoffen (Realkristalle, Gitterfehler, Gefüge) und Kunststoffen
- Eigenschaften der Werkstoffe (thermisch-mechanisches Verhalten, rheologisches Verhalten, elektrische Eigenschaften, Diffusion & Permeation)
- Modifikation der Eigenschaften (Legierungsbildung und Phasenumwandlung, Wärmebehandlungen und Mechanismen der Festigkeitssteigerung, Zusatzstoffe)
- Grundlagen der Werkstoffprüfung und -charakterisierung (Zugversuch für Metalle und Kunststoffe, Härteprüfung, Schlagzähigkeit, Wärmeformbeständigkeit, Thermische Analyse);

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Gastvorträge

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Weißbach: Werkstoffe und ihre Anwendungen;
- Bergmann: Werkstofftechnik 1+2
- Kaiser, Wolfgang: Kunststoffchemie für Ingenieure
- Ehrenstein, Gottfried: Polymer Werkstoffe
- Menges, Georg: Werkstoffkunde Kunststoffe

301 - Statistik und Stochastik

Empfohlenes Studiensemester: 3
Turnus: Wintersemester
Sprachen: Deutsch, Englisch
ECTS: 5
Prüfungsform: schrP oder ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Sarah Brockhaus (FK07)
Weitere Lehrende: Prof. Dr. Stephanie Thiemichen (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Die Studierenden sind vertraut mit einfachen mathematischen Problemstellungen und können diese analysieren und lösen. Insbesondere sind Grundkenntnisse in der Differential- und Integralrechnung auf Oberstufenniveau für elementare Funktionen vorhanden.

Lernziele

Fach- und Methodenkompetenz

Nach Abschluss des Moduls sind die Studierenden in der Lage:

Deskriptive Statistik:
- Datensätze systematisch zu erfassen, aufzubereiten und grafisch sowie tabellarisch darzustellen
- Kennzahlen wie Lage- und Streuungsmaße zu berechnen und zu interpretieren

Induktive Statistik:
- Grundbegriffe der Wahrscheinlichkeitsrechnung im Kontext statistischer Anwendungen zu verwenden
- Stichproben zu analysieren und Rückschlüsse auf Grundgesamtheiten zu ziehen
- Hypothesentests durchzuführen und korrekt zu interpretieren
- Konfidenzintervalle zu berechnen und zu interpretieren

Statistische Modellierung:
- Einfache lineare Regressionsanalysen durchzuführen und zu interpretieren
- Korrelationen zu erkennen und zu quantifizieren und von Kausalität zu unterscheiden

Datenanalyse:
- Datenquellen kritisch zu beurteilen
- geeignete Analysemethoden für konkrete Fragestellungen auszuwählen
- Datensätze mit Hilfe von Softwaretools (z.B. R, Python) zu analysieren

Kommunikation und Bewertung:
- Statistische Ergebnisse zu kommunizieren
- Aussagekraft, Grenzen und mögliche Fehlinterpretationen statistischer Analysen zu erkennen

Inhalt

Es werden folgende Themen behandelt:

Deskriptive Statistik:

- Merkmalstypen und Skalenniveaus
- Grafische Darstellungen (wie Balkendiagramm, Histogramm, Boxplot)
- Statistische Kennzahlen: Lagemaße (wie Mittelwert, Median, Quantile) und Streuungsmaße (wie Varianz, Standardabweichung, Interquartilsabstand)
- Bivariate Datenanalyse: Kontingeztabellen, Maße für den Zusammenhang (wie Korrelation), grafische Darstellungen (wie Scatter Plot)
- Problematik Korrelation und Kausalität

Grundlagen der Wahrscheinlichkeitsrechnung:

- Zufallsexperimente, Grundlagen der Kombinatorik (Fakultät und Binomialkoeffizient)
- Bedingte Wahrscheinlichkeiten und stochastische Unabhängigkeit
- Diskrete Zufallsvariablen und Verteilungen, Binomialverteilung, optional: Poisson-Verteilung, Gleichverteilung
- Stetige Zufallsvariablen und Verteilungen, Normalverteilung
- Erwartungswert und Varianz
- Optional: Mehrdimensionale Zufallsvariablen
- Optional: Gesetz der großen Zahlen, zentraler Grenzwertsatz

Induktive Statistik:

- Grundgesamtheit und Stichprobe
- Optional: Schätzprobleme, wie Schätzung von Erwartungswert und Varianz
- Konfidenzintervalle (z.B. für Erwartungswert, Anteilswert)
- Statistische Tests, t-Test/A-B-Testing, optional: Chi-Quadrat-Test

Statistische Modellierung:

- Einfache lineare Regressionsanalyse; optional: mehrdimensionale Regression

In den Übungen werden anhand von Aufgaben und Beispielen Verständnis und praktische Anwendung geübt. Die Studierenden verwenden dazu auch Computerwerkzeuge wie beispielsweise R oder Python.

Lehrmethoden und Lernformen

Tafel, Folien oder Beamer, Software-Tools für Statistik und Wahrscheinlichkeitsrechnung (beispielsweise R oder Python)

Verwendbarkeit des Moduls

Bachelor Digital Engineering, Bachelor Informatik und Design, Bachelor Geodata Science

Literatur

- Norbert Henze, Stochastik für Einsteiger, Vieweg
- Albrecht Irle, Wahrscheinlichkeitsheorie und Statistik, Teubner
- Ludwig Fahrmeir et.al., Statistik - der Weg zur Datenanalyse, Springer Spektrum
- Sandro Scheid und Stefanie Vogl, Data Science - Grundlagen, Methoden und Modelle der Statistik, Carl Hanser Verlag

302 - IT-Sicherheit und technischer Datenschutz

Empfohlenes Studiensemester: 3
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: ModA oder praP oder schrP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Peter Trapp (FK07), Prof. Dr. Thomas Schreck (FK07)
Weitere Lehrende: Prof. Dr. Erik Krempel (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Studierende lernen das Grundwissen der IT-Sicherheit sowie des technischen Datenschutzes. Im konkreten werden folgende Ziele adressiert:

- Studierende kennen die Grundlagen der IT-Sicherheit und des Datenschutzes.
- Studierende verstehen die Gefahren sowie Angriffe in der IT und können geeignete Maßnahmen vorschlagen.
- Studierende wenden die Grundzüge der Kryptographie praktisch an.
- Studierende verstehen die Grundlagen von digitalen Identitäten und der Zugriffskontrolle.
- Studierende können einen fehlerhaften Softwareentwicklungsprozess erkennen und diesen verbessern.
- Studierende verstehen die Grundprinzipien der DSGVO und können diese Systembezogen anwenden.
- Studierende können datenschutzfreundliche Softwarearchitekturen umsetzen.

Inhalt

- Grundlagen der IT-Sicherheit (z.B. CIA-Prinzip)
- Risiko(-management) und Gefahren
- Grundlagen der Kryptographie
- Digitale Identitäten
- Zugriffskontrolle
- Sichere Entwicklungs(-prozesse) (z.B. DevSecOps)
- Business Continuity Prozesse
- Grundlagen und Prinzipien des Datenschutzes
- Datenschutzkonzepte in der Softwareentwicklung

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Gastvorträge

Verwendbarkeit des Moduls

Bachelor Digital Engineering, Bachelor Geodata Science

Literatur

- IT-Sicherheit: Konzepte - Verfahren - Protokolle; Claudia Eckert; ISBN-13: 978-3-110-55158-7
- Informationssicherheit und Datenschutz: Handbuch für Praktiker und Begleitbuch zum T.I.S.P.; Secorvo; ISBN-13 : 978-3864905964
- Einführung in das Datenschutzrecht; Tinnefeld/Ehmann/Gerling; Oldenbourg 2005
- IT-Grundschutz-Handbuch; BSI; http://www.bsi.bund.de/gshb/index.htm
- aktuelle Literatur aus Internet (z.B. heise Security)

311 - CAD/Konstruktion

Empfohlenes Studiensemester: 3
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Jürgen Huber (FK03)
Weitere Lehrende: Prof. Dr.-Ing. Rainer Annast (FK03), Prof. Dr.-Ing. Mirko Langhorst (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Die Studierende können die räumliche Gestalt technischer Gebilde regel- und normengerecht darstellen bzw. aus technischen Zeichnungen deren Gestalt und Funktion ableiten. Sie sind in der Lage, Wechselwirkungen zwischen einzelnen Konstruktionselementen in einer Gesamtkonstruktion zu identifizieren und zu berücksichtigen. Sie besitzen die Grundlagen für die Entwicklung von Bauteilen, die zu moderaten Kosten fertigbar sind. Darüber hinaus beherrschen die Studierenden die Methodik der 2D- und 3D-Konstruktion unter Anwendung marktüblicher Software-Produkte.

Inhalt

- 3-Tafel-Projektion, axonometrische Darstellungen
- Grundlagen Technisches Zeichnen
- Oberflächen, Toleranzen, Passungen, Toleranzrechnungen
- Einblick in Form- und Lagetoleranzen
- Lastflussanalyse
- Konstruktionsskelett
- Einblick in die fertigungsgerechte Gestaltung von Bauteilen
- Darstellende Geometrie
- Grundlegende Funktionsweise von CAD-Systemen, vermittelt anhand praktischer Übungen

Lehrmethoden und Lernformen

Tafel, Flipchart, Folien, Beamer, Lehr-/Lernvideos, E-Learning

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Hoischen: Technisches Zeichnen, Cornelsen Verlag
- Roloff / Matek: Maschinenelemente, Springer Vieweg Verlag
- Sauer: Konstruktionselemente des Maschinenbaus 1 und 2, Springer Vieweg Verlag
- Niemann, Winter: Maschinenelemente Band I - III, Springer Vieweg Verlag
- Dubbel: Taschenbuch für den Maschinenbau Band I - III, Springer Verlag

313 - Cyber Physical Systems

Empfohlenes Studiensemester: 3
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: praP oder schrP oder mdlP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr.-Ing. Martin Orehek (FK07)
Weitere Lehrende: Prof. Dr. Stefan Wallentowitz (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Grundlegendes Verständnis der Informationsverarbeitung und Programmierung (z.B. erworben in Module: Computational Thinking, Softwareentwicklung). Grundlegende Kenntnisse der physikalischen Gesetzte und Zusammenhänge (z.B. erworben in Module: Physik, Technische Mechanik), der Funktionsweise elektronischer Bauteile (z.B. erworben im Modul: Elektrotechnik) und der Digitaltechnik (z.B. erworben im Modul: Computer Systems Fundamentals).

Lernziele

Fach- und Methodenkompetenz

Studierende lernen die Grundlagen von Cyber Physical Systems (CPS) kennen.

Einführung CPS: Die Studierenden

- erklären die folgenden Begriffe: Embedded Systems, Cyber Physical Systems
- stellen die Relevanz von CPS dar und identifizieren Anwendungsfelder
- benennen und erklären die Herausforderungen des Entwurfs von CPS

Spezifikation und Modellierung: Die Studierenden

- stellen heraus was ein Modell ist und warum beim Entwurf von CPS Modelle zum Einsatz kommen
- erklären die Definition eines Berechnungsmodells und wählen für konkrete Aufgabenstellungen geeignete Berechnungsmodelle aus
- erstellen für konkrete Problemstellungen geeignete Modelle
- simulieren und überprüfen das Verhalten von Modelle

Hardware Eingebetteter Systeme: Die Studierenden

- stellen die Hardwareschleife, die typischerweise bei Eingebetteten Systemen vorkommt, dar und erklären die Funktion der vorkommenden Elemente
- erklären die folgenden Begriffe und ordnen diese im technischen Kontext ein: Zeit- und Wertediskretisierung, Alias-Effekt, Quantisierungsabweichung, TDMA, CSMA/CD, CSMA/CA, PWM
- wählen für ein gegebenes System geeignete Schnittstellen zur Anbindung der physikalischen Umwelt aus
- verwenden gegebene HW-Plattformen, um im Rahmen von Projektaufgaben Systemlösungen zu entwickeln

Systemsoftware / Funktionssoftware: Die Studierenden

- erklären die Begriffe *Prozess* und *Thread* und grenzen dies voneinander ab
- erklären anhand selbst konstruierter Beispiele, die Probleme: *Prioritätsumkehr* und *Deadlock*
- erklären das *Priority Inheritance Protocoll* (PIP) und *Priority Ceiling Protocoll* (PCP) und wenden es an Beispielen an
- entwickeln für ein gegebenes CPS eine geeignete Softwarelösung
- verwenden gegebene SW-Plattformen und Frameworks, um im Rahmen von Projektaufgaben Systemlösungen zu entwickeln

Überfachliche Kompetenz

- Über Gruppenarbeiten werden kommunikative Kompetenzen gefördert.

Inhalt

- Einführung in Cyber Physical Systems: Motivation, Relevanz, Anwendungsfelder, Begriffe
- Eigenschaften CPS und Herausforderung bei deren Entwicklung
- Lösungsansätze für die Herausforderungen / Vorgehensmodelle
- Spezifikation und Modellierung (z.B. UML, Zustandsautomaten, Differentialgleichungen/Blockdiagramme, Stateflow/Simulink, ...)
- Hardware und Schnittstellen CPS (z.B. Standard-IO: Push-Pull / Open Output, PWM, I2C, UART, ADC, DAC, ...)
- Systemsoftware (z.B. Arduino, RTOS, ...)
- Bewertung und Validierung von Lösungsansätze
- Anwendung des Erlernten an einem konkreten Beispiel mit Hardware und Software

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Gastvorträge

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Marwedel Peter: *Eingebettete Systeme - Grundlagen Eingebetteter Systeme in Cyber-Physikalischen Systemen*. 2. Auflage, Springer Vieweg, Open Access, 2021.
- Lee A. E., Seshia A. S.: *Introduction to Embedded Systems, A Cyber-Physical Systems Approach*, Second Edition, MIT Press, ISBN 978-0-262-53381-2, 2017.
- Hüning F.: *Embedded Systems für IoT*. Berlin, Heidelberg: Springer Berlin Heidelberg.
- Barkalov A., Titarenko L., Mazurkiewicz M.: *Foundations of Embedded Systems*. Bd. 195. Cham: Springer International Publishing.
- Berns K., Köpper A., Schürmann B.: *Technische Grundlagen Eingebetteter Systeme: Elektronik, Systemtheorie, Komponenten und Analyse*. Wiesbaden: Springer Fachmedien Wiesbaden.
- Weitere Literatur wird im Rahmen der aktuellen Veranstaltung bekannt gegeben.

314 - Datenhaltung

Empfohlenes Studiensemester: 3
Turnus: Wintersemester
Sprachen: Deutsch, Englisch
ECTS: 5
Prüfungsform: ModA oder praP oder schrP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Stephanie Thiemichen (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Computational Thinking

Lernziele

Die Studierenden lernen verschiedene Methoden der Datenhaltung kennen und anzuwenden, um sie in ihrer späteren beruflichen Tätigkeit bei der Speicherung, Extraktion, Aggregation, Transformation und Filterung von Daten aus verschiedenen Datenquellen sinnvoll einsetzen zu können.

Fach- und Methodenkompetenz

Die Studierenden sind in der Lage

- die grundlegenden Konzepte hinter der Datenhaltung zu erläutern.
- Systeme entsprechend gegebener Anforderungen zu evaluieren.
- aktuelle Datenhaltungssysteme zu bedienen und in den Data Science Prozess einzubinden.

Überfachliche Kompetenz

- Teamarbeit: Die Studierenden bearbeiten Problemstellungen in Kleingruppen

Inhalt

- Grundlagen von Informationssystemen
- Entity-Relationship-Modell
- Relationale Datenbanken
- Einführung und Grundlagen SQL
- Sichten (Views) / Normalisierung
- Überblick XML / XPath / XQuery
- Einführung und Überblick nicht-relationale Datenbanken (NoSQL)
- Grundlagen verteilter Informationssysteme (z.B. Hadoop HDFS/HBase), Grundidee MapReduce
- Ausblick auf weitere Themen: z.B. Graph-Datenbanken, Geodatenhaltung, DVC, Vektordatenbanken

Im Praktikum werden anhand von Aufgaben und Beispielen Verständnis und praktische Anwendung geübt. Die Studierenden verwenden dazu auch SQL (z.B. MySQL) und zusätzliche Programmiersprachen (z.B. Python oder R).

Lehrmethoden und Lernformen

Beamer, Tafel, SQL Skripte und entsprechende Datenbankinstanzen, Tools zur grafischen ER-Modellierung (z.B. ERDPlus oder draw.io), Markdown zur Dokumentation.

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Alfons Kemper und André Eickler (2015): *Datenbanksysteme - Eine Einführung*. Vol. 10. Oldenbourg Verlag.
- Weitere Literatur wird im Rahmen der aktuellen Veranstaltung bekannt gegeben.

315 - Fertigungstechnik

Empfohlenes Studiensemester: 3
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (3SWS), Pra (1SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 45 Stunden Seminaristischer Unterricht, 15 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Matthias Reihle (FK03)
Weitere Lehrende: Prof. Dr. Mirko Langhorst (FK03), Prof. Dr. Henning Stoll (FK03), Prof. Dr. Tobias Hornfeck (FK03), Prof. Dr. Alexander Lindworsky (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

Werkstoffkunde

Lernziele

Fach- und Methodenkompetenz

Lernziel des Moduls ist die Fähigkeit zur Auswahl, Planung und Durchführung von Fertigungsverfahren unter Berücksichtigung des Zusammenwirkens von Werkstoff, Konstruktion und Fertigung. Die Studierenden sind nach erfolgreichem Bestehen des Moduls in der Lage, aus verschiedenen Verfahren die technisch und wirtschaftlich optimale Lösung zu ermitteln sowie die Auswirkungen auf die Bauteileigenschaften zu beurteilen.

Inhalt

- Hauptgruppen der Fertigungstechnik
- Aufgaben der Fertigungstechnik und Grundlagen zur industriellen Produktion
- Ausgewählte bzw. wichtige Fertigungsverfahren der Bereiche Urformen, Umformen, Trennen, Fügen und Beschichten
  - Urformen: Form- und Gießverfahren, Eigenschaften von Gussbauteilen und deren Beeinflussung, Kunststoffe & metallische Gusswerkstoffe
  - Umformen: Grundlegende Umformverfahren, Kenngrößen der Formänderung, Einfluss auf Bauteileigenschaften
  - Trennen: Zerspanungstechnik, Brennschneiden, Funkenerodieren
  - Fügen und Beschichten: Standard- und Sonderschweißverfahren, Schweißbarkeit eines Bauteils (Steuerung der Bauteileigenschaften), Beschichten mittels chemischer und physikalischer Gasphasenabscheidung (CVD & PVD)

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Gastvorträge

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Fritz, Alfred Herbert; Schulze, Günter: Fertigungstechnik
- Awiszus, Birgit; Bast, Jürgen; Dürr, Holger; Mayr, Peter: Grundlagen der Fertigungstechnik
- Klocke, Fritz; König, Wilfried: Fertigungsverfahren 1-5
- Michaeli, Walter; Hopmann, Christian: Einführung in die Kunststoffverarbeitung

402 - Künstliche Intelligenz

Empfohlenes Studiensemester: 4
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Übung
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. David Spieler (FK07)
Weitere Lehrende: Prof. Dr. Stephanie Thiemichen (FK07), Prof. Dr. Fabian Spanhel (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Eine moderne/aktuelle Programmiersprache (bestenfalls Python) gelehrt etwa in Computational Thinking

Lernziele

Die Studierenden lernen verschiedene Methoden der Künstlichen Intelligenz kennen und anzuwenden, um sie in ihrer späteren beruflichen Tätigkeit bei der Suche in Daten, Planung von Aktionen, Wissensrepräsentation / Inferenz als auch beim maschinellen Lernen sinnvoll einsetzen zu können.

Fach- und Methodenkompetenz

Die Studierenden sind in der Lage

- grundlegenden KI-Konzepte zu erläutern,
- einfache KI-Methoden selbst zu implementieren und
- komplexere KI-Methoden anzuwenden.

Sozialkompetenz

Teamarbeit: Die Studierenden bearbeiten Problemstellungen in Kleingruppen

Inhalt

- Tiefensuche
- Breitensuche
- A*
- Backtracking
- Adversariale Suche
- Logik und Inferenz
- Wissensrepräsentation und Inferenz (Beispiel RDF/OWL/SPARQL)

- Planen
- Unsicherheit
- Überwachtes Lernen
- Unüberwachtes Lernen
- Reinforcement Learning

Lehrmethoden und Lernformen

Seminaristischer Unterricht, Projektarbeit, Medien und Methoden: Beamer, Tafel, Jupyter Notebooks

Verwendbarkeit des Moduls

Bachelor Informatik und Design

Literatur

- Stuart J. Russell and Peter Norvig. *Künstliche Intelligenz. Ein moderner Ansatz*, 3., überarbeitete Edition, Pearson Studium, 2012.
- Lämmel U. and Cleve J., *Künstliche Intelligenz: Wissensverarbeitung - Neuronale Netze*, Carl Hanser Verlag GmbH & Company KG, 2020.
- James et.al. (2021): *An Introduction to Statistical Learning with Applications in R*, 2nd. Edition, Springer.
- Weitere Literatur wird im Rahmen der aktuellen Veranstaltung bekannt gegeben.

411 - Numerik

Empfohlenes Studiensemester: 4
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP oder ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Rainer Fischer (FK07)
Weitere Lehrende: Prof. Dr. Katina Warendorf (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Die Studierenden sind in der Lage,

- die wichtigsten numerischen Problemstellungen zu identifizieren.
- geeignete numerische Methoden und Algorithmen auszuwählen, fachgerecht zu implementieren, ihr Konstruktionsprinzip zu verstehen, ihre Grenzen zu kennen, sie sicher anzuwenden und auf spezielle Problemstellungen anzupassen.
- die Ursachen für das Versagen eines Algorithmus zu analysieren und fachgerecht zu beheben.

Inhalt

Einführung in den Entwurf und die Analyse von numerischen Methoden und Algorithmen zur Lösung mathematischer Probleme in Wissenschaft und Technik. Behandelt wird eine Auswahl aus folgenden Themen:

- Fehleranalyse, Stabilität von Algorithmen
- Lösung linearer Gleichungssysteme
- Ausgleichsrechnung
- Lösung nichtlinearer Gleichungssysteme
- Polynominterpolation (auch mit abschnittsweise definierten Polynomen)
- Numerische Differentiation und Integration
- Lösen einfacher Anfangswertprobleme

In dem begleitenden Praktikum werden Anwendungsaufgaben gelöst. Dazu wird ein Programmsystem zur Lösung von Aufgaben des wissenschaftlichen Rechnens verwendet, z.B. Matlab oder Python.

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Gastvorträge

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Michael T. Heath, Scientific Computing: An Introductory Survey, McGraw-Hill Higher Education, ISBN 978-0071244893
- Timothy Sauer, Numerical Analysis, Pearson, ISBN 0-321-46135-5
- Charles F. Van Loan, Introduction to Scientific Computing: A Matrix-Vector Approach Using Matlab, Pearson, ISBN 0-13-125444-8
- Wolfgang Preuß and Günter Wenisch (Hrsg.), Lehr- und Übungsbuch Numerische Mathematik, Fachbuchverlag Leipzig, ISBN 3-446-21375-9
- Cleve B. Moler, Numerical Computing with MATLAB, Society for Industrial Mathematics, ISBN 978-0898715606
- Svein Linge and Hans Petter Langtangen, Programming for Computations Python, Springer, ISBN 978-3-030-16876-6
- Alfio Quarteroni, Fausto Saleri, and Paula Gervasio: Scientific Computing with Matlab and Octave, Springer, ISBN 978-3-642-45367-0

413 - Smart Systems

Empfohlenes Studiensemester: 4
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: praP oder schrP oder mdlP oder ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr.-Ing. Martin Orehek (FK07)
Weitere Lehrende: Prof. Dr. Stefan Wallentowitz (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Grundlegendes Verständnis der Informationsverarbeitung, der Programmierung und der Softwareentwicklung (z.B. erworben in Module: Computational Thinking, Software Engineering). Grundlegende Kenntnisse einer höheren Programmiersprache (z.B. erworben im Modul: Softwareentwicklung) und der Netzwerkkommunikation (z.B. erworben im Modul: Computer Systems Fundamentals).

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Lernziele

Fach- und Methodenkompetenz

Studierende lernen die Grundlagen zu Smarte Systeme kennen. Sie lernen wie konkrete Smarte Systeme aufgebaut sind und die Smartness realisiert werden kann.

Einführung - Digitale Transformation und Digitalisierung: Die Studierenden

- erklären folgende Begriffe und können diese voneinander abgrenzen: Digitale Transformation, Digitalisierung, Internet der Dinge (IoT), Industrie 4.0, Big Data, Machine Learning
- stellen dar wir *Smart Systems* ein Teil der Digitalen Transformation sind

Eigenschaften Smarter Systeme: Die Studierenden

- erklären wie eine systematische Literaturrecherche durchgeführt wird
- führen selbst eine Literaturrecherche zu einem gegebenen Thema durch
- präsentieren welche Konzepte zur Charakterisierung von Systemen genutzt werden können
- beschreiben wie *Smartness* in Domänen definiert wird (z.B. Smart-City, -Home, -Industry, -Service-System, ...)
- beschreiben welche Eigenschaften Smarte Systeme besitzen (Infos aus der Literaturrecherche)

IoT Architekturen - Anwendungen und Herausforderungen: Die Studierenden

- beschreiben die unterschiedlichen IoT Architekturen aus wissenschaftlichen Veröffentlichungen, analysieren Gemeinsamkeiten und Unterschiede
- erklären und differenzieren folgende Begriffe: *Cloud-*, *Fog-*, *Edge-Computing*
- beschreiben verschiedene IoT Anwendungsbereiche und die darin verfolgten Ziele
- benennen die Herausforderungen bei der Umsetzung von IoT Anwendungen

Netzwerke - Vertiefung: Die Studierenden

- beschreiben und erklären die Aufgaben und grundlegenden Funktionen der Transportschicht
- analysieren die Anforderungen von Aufgabenstellungen und wählen geeignete Transportprotokolle aus
- beschreiben die Aufgaben der Netzwerkschicht und erklären die grundlegenden Funktionen der verwendeten Protokolle

IoT Protokolle: Die Studierenden

- beschreiben etablierten IoT Protokolle und erklären deren Funktionsweisen und Eigenschaften
- analysieren Aufgabenstellungen, wählen dafür geeignete Protokolle aus und begründen die Wahl nachvollziehbar
- vergleichen die Protokolle miteinander

Einführung - Algorithmen: Die Studierenden

- erklären was ein Algorithmus im Kontext der Informationsverarbeitung ist
- beschreiben die Kerneigenschaften von Algorithmen
- kennen, interpretieren und nutzten Beschreibungsformen für Algorithmen
- erklären die folgenden Begriffe und grenzen diese voneinander ab: Beschreibungs-, Laufzeit-, Speicherkomplexität, Random Access Machine (RAM), Asymptotische Analyse, Landau-Notation, O-Notation
- untersuchen, bewerten und vergleichen Algorithmen unter den Aspekten: Komplexität und Korrektheit
- nennen wichtige Laufzeitklassen und stellen diese in Bezug
- berechnen die Laufzeitklasse für gegebene Laufzeitfunktionen und wenden dabei die Rechenregeln der O-Notation korrekt an
- analysieren gegebene Algorithmen, bestimmen dafür die Worst-Case-Laufzeit und berechnen die entsprechende Laufzeitklasse

Begleitendes Anwendungsbeispiel: Die Studierenden

- analysieren eine gegebene Aufgabenstellung und leiten daraus die technischen Anforderungen für ein System ab
- wählen aus einer Menge von existierenden Technologien, die für die Aufgabenstellung optimale Lösung aus
- realisieren eine *Minimum Viable Product (MVP)* für ein System

Überfachliche Kompetenz

- Über Gruppenarbeiten werden kommunikative Kompetenzen gefördert.

Inhalt

- Systematische Literaturrecherche und Analyse der verfügbaren Veröffentlichungen zu Themen wie z.B.: Smart-X, IoT Architekturen
- IoT Referenzarchitektur der Eclipse Foundation
- Netzwerke: TCP/UDP, IPv4/IPv6, Adressierung, Routing, Routing Algorithmen, ...
- IoT Protokolle: MQTT, MQTT-SN, CoAP, LwM2M, Sparkplug, ...
- Algorithmen: Beschreibungsformen (z.B. UML Aktivitätsdiagramme, Pseudocode), konkrete Beispiele je nach Projektaufgaben

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Gastvorträge

Literatur

- Borgmeier A., Grohmann A., Gross S. F., *Smart Services und Internet der Dinge: Geschäftsmodelle, Umsetzung und Best Practices: Industrie 4.0, Big Data, Machine Learning, Blockchain, kollaborative Ökosysteme, Human Centricity*, 2., Vollständig überarbeitete Auflage. München: Hanser, 2022
- Romero M., Guédria W., Panetto H., Barafort B., *Towards a Characterisation of Smart Systems: A Systematic Literature Review*, Computers in Industry, Bd. 120, S. 103224, Sep. 2020
- Atlam H. F., Walters R. J., Wills G. B., *Internet of Things: State-of-the-art, Challenges, Applications, and Open Issues*, IJICR, Bd. 9, Nr. 3, S. 928–938, Sep. 2018
- Sethi P., Sarangi S. R., *Internet of Things: Architectures, Protocols, and Applications*, Journal of Electrical and Computer Engineering, Bd. 2017, S. 1–25, 2017
- Desbiens F., *Building Enterprise IoT Solutions with Eclipse IoT Technologies: An Open Source Approach to Edge Computing*. Berkeley, CA: Apress, 2023
- Weitere Literatur wird im Rahmen der aktuellen Veranstaltung bekannt gegeben.

414 - Messtechnik/Sensoren

Empfohlenes Studiensemester: 4
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: praP oder schrP oder mdlP oder ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr.-Ing. Frank Palme (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Studierende lernen die Grundlagen der Messtechnik und Sensoren mit den Lernzielen, dass sie

- die Grundlagen des Messens und der Anwendungsgebiete von Sensoren basierend auf deren physikalischen Funktionsprinzipien erläutern.
- messtechnische Grundstrukturen und deren spezifische Einsatzgebiete benennen.
- die Fähigkeit erlangen Messverfahren zu planen, geeignete Sensoren, Filter, Verstärker und analoge/digitale Auswertemethoden auszuwählen, sowie Messaufgaben inklusive Kalibrierung und Messunsicherheitsbetrachtungen durchzuführen.
- die Kompetenz entwickeln die Messqualität zu beurteilen und Messergebnisse zu interpretieren.
- Messsysteme dabei insbesondere im Bewusstsein von Nachhaltigkeitszielen einsetzen.

Inhalt

- Theoretische Grundlagen der Messtechnik, fundamentaler Sensoren und der zugehörigen sensorspezifischen Messverfahren
- Analoge und digitale Methoden zur Kalibrierung, Erfassung, Verarbeitung, Übertragung, Auswertung und Darstellung von Messsignalen
- Eigenschaften und Kenngrößen von Messsystemen:
  - statische Kenngrößen: Messbereich, Empfindlichkeit, Kennlinie, Messunsicherheit, Fehlerfortpflanzung
  - dynamische Kenngrößen: Übertragungsverhalten im Zeit- und Frequenzbereich, Übertragungsfunktionen, Filterung
- Auslegung der Messkette zur Erfassung fundamentaler mechanischer, thermischer, optischer und elektrischer Messgrößen (wie Kraft, Weg, Dehnung, Druck, Temperatur, Spannung, Strom, Leistung, Lichtstärke, Zeit, Frequenz)

- Entwurf, Simulation und Anwendung von Messsystemen und Sensoren anhand praxisorientierter Beispiele und Übungen

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, digitale Lehrmedien

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Profos, P., Pfeifer, T. (Hrsg.): Handbuch der industriellen Messtechnik. Oldenbourg, München (1994)
- Gevatter, H.J., Grünhaupt, U. (Hrsg.): Handbuch der Mess- und Automatisierungstechnik in der Produktion. Springer, Berlin Heidelberg New York (2006)
- Bergmann, K.: Einführung in die elektrische Messtechnik. Vieweg+Teubner, Wiesbaden (2008)

415 - Thermodynamik & Fluidmechanik

Empfohlenes Studiensemester: 4
Turnus: Sommersemester
Sprachen: Deutsch, Englisch
ECTS: 6
Prüfungsform: schrP, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (5SWS)
Gesamter Workload: 180 Stunden
Präsenzzeit: 75 Stunden Seminaristischer Unterricht
Selbststudium: 105 Stunden
Modulverantwortung: Prof. Dr. Andreas Gubner (FK03)
Weitere Lehrende: Prof. Dr. Monika Mühlbauer (FK04)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Lernziele

Fach- und Methodenkompetenz

Studierende lernen die methodischen und fachlichen Qualifikationen zur thermodynamischen Analyse technischer Systeme. Konkret werden folgende Ziele adressiert:

- Studierende beschreiben die Zustandsänderungen und die Energieumwandlungsvorgänge von Gasen und Flüssigkeiten aufbauend auf der physikalischen Grundlagenvorlesung.
- Studierende verwenden die Fachsprache der Thermodynamik.
- Studierende arbeiten die thermodynamische Prozesse in technischen Systemen heraus.
- Studierende beschreiben und erkläreb die wesentlichen Mechanismen der Wärmeübertragung und wenden diese in Berechnungen an.
- Studierende führen die Berechnung bei einfachem Stoffverhalten durch.
- Studierende sind mit der Modellbildung der technischen Strömungslehre und mit den Elementaren vertraut.
- Studierende wenden die theoretischen Grundlagen zur Lösung konkreter Aufgaben an.

Inhalt

- Grundbegriffe der Thermodynamik und Wärmeübertragung: System, Zustand, Zustandsgrößen, Gleichgewicht, Zustandsänderung, Prozess
- Erster Hauptsatz: Energieformen, geschlossene und offene, stationäre Systeme, wichtige Anwendungen
- Verhalten idealer Gase: thermische und kalorische Zustandsgleichung, Mischungen, einfache Zustandsänderungen

- Zweiter Hauptsatz: Formulierungen und Aussagen, Entropie und Entropiebilanz, Anwendungen, Prozesse in Apparaten und Maschinen
- Kreisprozesse mit idealen Gasen
- Mehrphasensysteme reiner Stoffe
- Grundlagen der stationären Wärmeleitung, des konvektiven Wärmeübergangs (erzwungene und freie Konvektion) und der Grundlagen der Wärmestrahlung
- Wärmedurchgang an einfachen Geometrien
- Einführung in die Strömungsmechanik: Kontinuumsannahme und Strömungskinematik
- Grundgleichungen der Strömungsmechanik: Bilanzen der Energie-, Massen- und Impulserhaltung
- Widerstand umströmter Körper
- Rohrströmungen

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Flipcharts

Literatur

- Cerbe, G.; Wilhelms, G.: Technische Thermodynamik. Theoretische Grundlagen und praktische Anwendungen. Hanser.
- Langeheinecke, K.; Jany, P.; Thieleke, G.: Thermodynamik für Ingenieure. Springer Vieweg.
- Baehr, H.D.; Kabelac, S.: Thermodynamik. Springer.
- Böckh, P. v; Wetzel, T.: Wärmeübertragung. Grundlagen und Praxis. Springer.
- Herwig, H.; Moschallski, A.: Wärmeübertragung. Springer Vieweg.
- VDI-Gesellschaft Verfahrenstechnik und Chemieingenieurwesen (Hrsg.): VDI-Wärmeatlas. Springer.
- Cengel, Y.A.; Boles, M.A.: Thermodynamics. An Engineering Approach. Mc Graw Hill.
- National Institute of Standards and Technology: Reference Fluid Thermodynamic and Transport Properties – REFPROP. User's Guide.
- Böswirth, Bschorer: Technische Strömungsmechanik, Vieweg+Teubner.
- Böckh: Fluidmechanik, Vieweg+Teubner Hakenesch: Strömungsmechanik für Dummies, WILEY.

511 - Betriebswirtschaftslehre und Projektmanagement

Empfohlenes Studiensemester: 5
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP oder ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: SU (2SWS), Ü (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Übung
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Julia Eiche (FK03)
Weitere Lehrende: Prof. Dr. Maria Fritz (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

keine

Lernziele

Fach- und Methodenkompetenz

Die Studierenden können die wesentlichen betriebswirtschaftlichen Prozesse im Zusammenhang mit der Leistungserstellung und -verwertung beschreiben. Sie erläutern die Grundzüge der strategischen Ausrichtung von Unternehmen. Darüber hinaus beschreiben und erläutern sie die Besonderheiten im Entrepreneurship, insbesondere bei Start-ups, und analysieren und diskutieren die damit verbundenen Herausforderungen. Sie können die Grundlagen des Kostenmanagements im Unternehmen beschreiben, Controlling-Instrumente des externen und internen Rechnungswesens interpretieren, in geeigneten Fallstudien anwenden sowie zentrale Kennzahlen berechnen und vergleichen. Sie beschreiben die Disziplinen der Betriebswirtschaftslehre, erläutern deren Zusammenhänge und wenden das Gelernte in praxisbezogenen Fallbeispielen mit Bezug zu Industrieunternehmen an. Die Studierenden sind in der Lage, die Instrumente des klassischen und agilen Projektmanagements zu analysieren, deren Eignung für unterschiedliche Projektsituationen zu bewerten und geeignete Methoden anzuwenden. Sie verstehen, analysieren und diskutieren betriebswirtschaftliche Aspekte der aktuellen Wirtschaftspresse, insbesondere in Bezug auf Industrieunternehmen. Darüber hinaus verstehen, analysieren und diskutieren sie die grundlegenden Rahmenbedingungen und neuen Herausforderungen wirtschaftlichen Handelns in Bezug auf ökonomische, rechtliche, technologische, ökologische und gesellschaftliche Aspekte.

Inhalt

Grundbegriffe BWL, betrieblicher Umsatzprozess, Analyse und Gestaltung grundlegender Unternehmensstrategien Dimensionen Entrepreneurship Betriebswirtschaftliche Disziplinen (Forschung und Entwicklung, Materialwirtschaft, Produktion, Marketing, Vertrieb, Investition, Finanzierung) Kostenrechnung und Kostenmanagement, betriebliche Wertschöpfung Branchenrelevante Markt- und Unternehmensentwicklungen (z.B. aktuelle Wirtschaftspresse, Fallstudien, Geschäftsberichte, Praxisbeispiele, etc.), Grundbegriffe Projekte und Projektmanagement, ausgewählte Instrumente des klassischen und agilen Projektmanagement Unternehmerisches Planspiel

Lehrmethoden und Lernformen

Tafel, Beamer, Lehr-/Lernvideos, Videokonferenzen, Planspiel, evtl. Gastvorträge

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Thommen, J.-P. und Achleitner, A.-K.: Allgemeine Betriebswirtschaftslehre, Springer Gabler Verlag, akt. Auflage

513 - Ingenieurpraktikum mit Praxisseminar

Empfohlenes Studiensemester: 5
Turnus: Wintersemester
Sprachen: Deutsch, Englisch
ECTS: 25
Prüfungsform: ModA, weitere Angaben siehe SPO und Studienplan
Lehrform und SWS: Praktikum
Gesamter Workload: 20 Wochen in Vollzeit
Präsenzzeit: keine
Selbststudium: gesamt
Modulverantwortung: Prof. Dr.-Ing. Martin Orehek (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Formale Voraussetzung entsprechend der SPO: mindestens 90 ECTS-Kreditpunkte aus den ersten vier Studiensemestern

Lernziele

Ziele des betrieblichen Praktikums sind:

- Anwendung der im bisherigen Studium erworbenen Fähigkeiten in einem professionellen Kontext
- Feedback außerhalb des akademischen Umfelds zu den eigenen fachlichen und sozialen Kompetenzen
- Anregungen zur künftigen Gestaltung des Studiums und des Berufsstarts
- berufliche Vernetzung außerhalb der Hochschule

Die Studierenden können ihre zuvor im akademischen Feld erworbenen Fähigkeiten innerhalb der industriellen Praxis anwenden sowie ihre berufliche Orientierung und die Anforderungen der betrieblichen Praxis erkennen und in der Bedeutung für den eigenen Lernprozess einschätzen. Die Studierenden sind in der Lage, die Unterschiede der Arbeitsmethodik in der, industriellen Praxis gegenüber der wissenschaftlichen Arbeitsmethodik zu erkennen. Sie können die Gründe für die unterschiedlichen Vorgehensweisen nachvollziehen und sind in der Lage gemeinsame Bezugspunkte zu identifizieren. Bei Praktika in international tätigen Unternehmen oder direkt im Rahmen eines Auslandspraktikums stärken die Studierenden ihre Fremdsprachenkompetenz. Durch die heutzutage übliche Arbeit in Teams und die Einordnung in die Organisationsstruktur des Unternehmens werden die Soft Skills und sozialen Kompetenzen der Studierenden gestärkt. Die verantwortungsvolle Mitarbeit in Unternehmen, die sich alle täglich Ihrer gesellschaftlichen und sozialen Verantwortung stellen müssen (und das oft in Ihren Leitsätzen auch bereits formuliert haben), überträgt sich auch auf die im Praktikum engagierten Studierenden. Im Praxisseminar üben die Studierenden die Erstellung von Berichten mit ingenieurwissenschaftlicher Form und Inhalt ein.

Inhalt

Im praktischen Studiensemester sollen Studierende in die Tätigkeiten von Digital Engineers anhand konkreter Aufgabenstellungen eingeführt werden. Die Aufgabenstellungen sollen für Digital Engineers typischen Gebieten stammen:

- Computer-based Engineering
- Produkt(ions)design und -planung
- Softwareentwicklung im Industriekontext
- Forschung und Vorentwicklung
- Technische Projektleitung und -management
- Prüfung, Abnahme, Qualitätswesen
- Technischer Vertrieb

Im Praxisseminar verfassen Studierende einen Bericht mit ingenieurwissenschaftlicher Form und Inhalt zu einem Thema aus Ihrer Tätigkeit im Praxissemester.

Lehrmethoden und Lernformen

Lernen durch Erfahrung

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

keine

601 - Cloud Computing

Empfohlenes Studiensemester: 6
Turnus: Sommersemester
Sprachen: Deutsch, Englisch
ECTS: 5
Prüfungsform: ModA, weitere Angaben siehe Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Johannes Ebke (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Kenntnisse in mindestens einer höheren Programmiersprache

Lernziele

Fach- und Methodenkompetenz

- Studierende kennen die Konzepte und Prinzipien des Cloud Computing.
- Studierende können Cloud-Technologien und Cloud-Dienste einordnen und abgrenzen.
- Studierende kombinieren zielgerichtet Cloud-Dienste, um gegebene Problemstellungen zu lösen.
- Studierende bewerten, inwieweit gegebene Cloud-Architekturen die Erreichung von Softwarequalitätskriterien ermöglichen.
- Studierende kennen grundlegende Elemente der Sicherheit in Cloud-Infrastrukturen und können ihre Verantwortlichkeiten beschreiben.
- Studierende können einfache Cloud-Systeme überwachen und betreiben.

Inhalt

- Grundlagen des Cloud Computing (z.B. Virtualisierung, Kommunikation)
- Automatisierte Infrastrukturbereitstellung und Provisionierung (Infrastructure as a Service, Infrastructure as Code)
- Cluster Scheduling & Orchestration
- Cloud-Architekturen und Programmiermodelle (Serverless Computing)
- Monitoring und DevOps
- Sicherheitsaspekte der Cloud-Nutzung und Sicherheitsmodelle

Lehrmethoden und Lernformen

Whiteboard, Beamer, Digitale Quellen und Lernplattformen sowie praktische Aufgaben unter Nutzung eines oder mehrerer öffentlicher Cloud-Provider

Verwendbarkeit des Moduls

Bachelor Digital Engineering, Bachelor Geodata Science, Wahlpflichtmodul Bachelor Informatik und Design

Literatur

- Aktuelle Online-Quellen und Dokumentationen
- Boris Scholl et. al.: Cloud Native (2019), ISBN: 978-1-49205-382-8
- John Arundel: Cloud Native DevOps mit Kubernetes (2019), ISBN: 978-3-86490-698-5
- Eberhard Wolff: Das Microservices-Praxisbuch: Grundlagen, Konzepte und Rezepte (2018), ISBN 978-3-86490-526-1
- Chris Dodson: Practical Cloud Security: a guide for secure design and deployment (2019), ISBN 978-1-492-03751-4
- Betsy Beyer et al.: Site reliability engineering: How Google runs production systems. (2016) ISBN 978-1-491-92912-4

611 - Visual Computing

Empfohlenes Studiensemester: 6
Turnus: Sommersemester
Sprachen: Deutsch, Englisch
ECTS: 5
Prüfungsform: ModA, weitere Angaben siehe Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Markus Friedrich (FK07)

Empfohlene Vorkenntnisse für die Teilnahme

Kenntnisse in mindestens einer höheren Programmiersprache

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Lernziele

Fach- und Methodenkompetenz

Studierende lernen die Anwendung der grundlegenden Konzepte des Visual Computings: Visual Computing bedeutet aus Bildern Informationen zu gewinnen und Informationen in Bilder zu verwandeln. Die Darstellung von Informationen ist sowohl mit 2D Bildern als auch in 3D durch virtuelle Realität oder Mixed Reality möglich.

- Studierende kennen grundlegende Prinzipien der Verarbeitung von digitalen Bildern.
- Studierende kennen grundlegende Prinzipien der Computergrafik.
- Studierende kennen grundlegende Prinzipien der Informationsvisualisierung.
- Studierende kennen grundlegende Prinzipien der virtuellen Realität und der Mixed Reality.
- Studierende können Anwendungsmöglichkeiten von Künstlicher Intelligenz und maschinellem Lernen mit digitalen Bildern beschreiben.

Inhalt

- Einführung in die Grundlagen der Bildverarbeitung:
  - Digitale Bilddaten und Sensoren
  - Farbwarnehmung und Farbmodelle
  - Digitale Filter
  - Bildmerkmale
  - Maschinelles Sehen
- Einführung in die Grundlagen der Computergrafik:

  - Die Rendering Pipeline
  - Geometrische Modellierung
  - Koordinatensysteme, Transformationen und Projektionen
  - Beleuchtung und Schattierung
  - Texture Mapping
- Einführung in die Grundlagen der Informationsvisualisierung
  - Datentypen und Visualisierungstechniken
  - Prinzipien der effektiven Gestaltung
  - Interaktive Visualisierung und Benutzererfahrung
  - Scientific Visualization
  - Ethische Aspekte
- Einführung in die Grundlagen der Mixed Reality
  - Das Mixed Reality Spektrum
  - Technische Grundlagen (Tracking, Erfassung der Umgebung)
  - Aktuelle Mixed Reality-Geräte und -Technologien
  - Entwurfsprinzipien und -prozesse
  - Fallstudien und Anwendungen

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, Lehr-/Lernvideos, Gastvorträge, projektbasiertes Lernen, virtuelle Lehrräume

Literatur

- Nischwitz, Fischer, Haberäcker, Socher; Bildverarbeitung, Springer Vieweg, 2020
- Szeliski; Computer Vision: Algorithms and Applications, Springer, 2022
- Nischwitz, Fischer, Haberäcker, Socher; Computergrafik, Springer Vieweg, 2019
- Spence; Information Visualization: An Introduction, Springer, 2014
- Dörner, Broll, Grimm, Jung; Virtual und Augmented Reality (VR/AR); Springer Vieweg, 2019

612 - Digitale Signalverarbeitung

Empfohlenes Studiensemester: 6
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: praP oder schrP oder mdlP oder ModA, weitere Angaben siehe Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr.-Ing. Frank Palme (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

Grundlagen der Messtechnik und der Elektrotechnik

Lernziele

Fach- und Methodenkompetenz

Studierende lernen Theorie und Praxis der Digitalen Signalverarbeitung mit den Lernzielen, dass sie

- aktuelle Anwendungsgebiete und Lösungsansätze mit Digitaler Signalverarbeitung (DSP) aufzeigen und erklären.
- hierzu fundamentale Eigenschaften, Methoden, Algorithmen und Effekte zeitdiskreter digitaler Systeme identifzieren und formulieren.
- aktuelle Messtechnik/DSP-Hardware und -Software kennen und anhand gegebener Anforderungen unter Anwendung systematischer Entwurfsmethodik auslegen können.
- DSP-Systeme nicht nur bedienen, sondern auch beurteilen und vorteilhaft zur Problemlösung einsetzen können, insbesondere in den Bereichen Signal-/Spektralanalyse, Umweltsensorik und Autonome Systeme.
- die Kompetenz entwickeln, die Datenqualität zu beurteilen, Messergebnisse zu interpretieren, und Störeinflüsse zu identifizieren.
- DSP-Systeme dabei insbesondere im Bewusstsein von Sicherheits-, Datenschutz-, Ethik- und Nachhaltigkeitskriterien einsetzen.

Inhalt

- Methoden der Digitalen Signalverarbeitung: Zeitkontinuierliche und zeitdiskrete Systeme, Abtastung und Digitalisierung, Abtasttheorem, Leakage-Effekt, Fensterung, Faltung, Spektraltransformationen, Übertragungsfunktionen, Frequenz- und Systemverhalten, Digitale Filter, Korrelation
- Systeme zur Digitalen Signalverarbeitung: Aufbau, Kenngrößen, Komponenten, Konfiguration, Programmierung, Kalibrierung
- Hardware: Aufbau und Funktion von DSP-Systemen und deren Komponenten, Kriterien für Auswahl und Entwurf von DSP-Systemen (wie Genauigkeiten, Datenqualität, Echtzeitanforderungen, statistische Betrachtungen)

- Software: Programmierung von DSP-Systemen mit Hilfe gängiger Software-Tools/-Bibliotheken (wie LabVIEW, Matlab oder in Hochsprache): Datenstrukturen, Timing, Messwerterfassung, Algorithmen, Speicherverwaltung, virtuelle Bedien- und Anzeige-elemente, Visualisierung, Fehlerbehandlung. Spezielle DSP-Verfahren und Anknüpfung an Methoden der Bildverarbeitung und Künstlichen Intelligenz / Machine Learning
- Entwurf, Simulation und Anwendung von DSP-Systemen anhand praxisorientierter Beispiele und Übungen

Lehrmethoden und Lernformen

Tafel, Folien, Beamer, digitale Lehrmedien

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Kammeyer, K., Kroschel, K.: Digitale Signalverarbeitung. Vieweg+Teubner, Wiesbaden (2002)
- Meyer, M.: Signalverarbeitung. Springer Vieweg, Wiesbaden (2021)
- Jamal, R., Hagestedt, A.: LabVIEW - Das Grundlagenbuch. Pearson (2004)

613 - Modellbildung und Simulation

Empfohlenes Studiensemester: 6
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 10
Prüfungsform: ModA, weitere Angaben siehe Studienplan
Lehrform und SWS: SU (4SWS), Ü (4SWS)
Gesamter Workload: 300 Stunden
Präsenzzeit: 60 Stunden Seminaristischer Unterricht, 60 Stunden Übung
Selbststudium: 180 Stunden
Modulverantwortung: Prof. Dr. Markus Gitterle (FK03)
Weitere Lehrende: Prof. Dr. Wolfgang Högele (FK07), Prof. Dr. Björn Kniesner (FK03), Prof. Dr. Gerta Köster (FK07), Prof. Dr. Norbert Nitzsche (FK03), Prof. Dr. Martin Orehek (FK07), Prof. Dr. Henning Stoll (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

Physik, Technische Mechanik, Werkstoffkunde, Thermodynamik und Fluidmechanik, Numerik

Lernziele

Fach- und Methodenkompetenz

Mit Abschluss der Modulveranstaltung

- kennen Studierende unterschiedliche Vorgehensweisen zum Beschreiben von technischen Zusammenhängen in Modellen.
- sind Studierende in der Lage Modelle zu realisieren.
- sind Studierende sensibilisiert hinsichtlich Anwendbarkeit und Grenzen von Modellen, auch der ihnen bereits bekannten Modelle aus dem Ingenieurwesen.
- sind Studierende in der Lage, ihnen bereits bekannte Modelle aus dem Ingenieurwesen mit Hilfe des Computers zu simulieren / numerisch zu lösen.
- sind Studierende in der Lage, Simulationsergebnisse in ausgewählten Feldern des Ingenieurwesens zu beurteilen.
- wissen Studierende beispielhaft von der Abhängigkeit ausgewählter Modelle und von der Notwendigkeit, diese zu koppeln.

Überfachliche Kompetenz

- Über Gruppenarbeiten werden kommunikative Kompetenzen gefördert.

Inhalt

- Methodische Grundlagen der Modellbildung und Simulation von Systemen:
  - Beobachtung
  - Abstraktion zum Modell

  - Diskretisierung
  - Algorithmus
  - Simulation
  - Verifikation und Validierung
- Numerische Festkörpermechanik, Finite Elemente für aufgewählte mechanische Modelle, theoretische Grundlagen und angewandte Simulationen mit kommerziellem Programm (z.B. Ansys).
- Numerische Strömungsmechanikmechanik, Finite Volumen zur Lösung von Strömungmodellen, theoretische Grundlagen und angewandte Simulationen mit kommerziellem Programm (z.B. Ansys).
- Simulationen für Werkstoff- und fertigungsgerechte Gestaltung in der Kunststofftechnik, Modellierung, Besonderheiten und Kopplung zur Strukturmechanik (z.B. Moldex3D).
- Simulationsmodelle technischer Prozesse (z.B. Mehrkörpersystem) als Grundlage der Regelungstechnik, theoretische Grundlagen und angewendete Simulationen mit kommerziellen Werkzeugen (z.B. Matlab/Simulink).

Lehrmethoden und Lernformen

Tafel, Folien, Beamer

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Bungartz, H., Zimmer, S., Buchholz, M., Pflüger,D.: Modellbildung und Simulation, Springer, 2013.
- Gross, D., Hauger, W., Wriggers, P.: Technische Mechanik 4, Hydromechanik, Elemente der Höheren Mechanik, Numerische Methoden, Springer, 2018.
- Belytschko, T., Liu, W.K., Moran, B., Elkhodary, K.I.: Nonlinear finite elements for continua and structure, Wiley, 2014.
- Ferziger, J., Peric, M.: Numerische Strömungsmechanik, Springer Verlag, 2008.

614 - Regelungstechnik

Empfohlenes Studiensemester: 6
Turnus: Sommersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP oder ModA, weitere Angaben siehe Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Norbert Nitzsche (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

- Mathematik 1 und 2
- Elektrotechnik
- Technische Mechanik

Lernziele

Fach- und Methodenkompetenz

Die Studierenden sind in der Lage, Regelungsprobleme mathematisch zu beschreiben und erläutern die Grundlagen des analytischen Reglerentwurfs auf Basis von Übertragungsfunktionsmodellen und Zustandsraummodellen. Dabei erlernen sie numerische Werkzeuge und Computer Algebra Systemen anzuwenden. Die Studierenden verstehen die verschiedenen Zielkonflikte beim Reglerentwurf und können auf dieser Grundlage an bestehenden Regler experimentell Feinabstimmungen vornehmen. Außerdem können die Studierenden erklären, wie Regelgesetze digital zu implementieren sind.

Inhalt

- Überführen bekannter physikalischer Zusammenhänge in Differentialgleichungs-, Übertragungsfunktions- und Zustandsraummodelle (zeitkontinuierlich und zeitdiskret)
- Klassifikation von Übertragungsverhalten
- Blockschaltbildalgebra
- Reglerentwurf für P-, PI-, PD- und PID-Regler mit Methoden auf Basis von Übertragungsfunktionen (Polvorgabe, Pol-Nullstellen-Kompensation, Frequenzgangmethoden, Wurzelortskurven)
- Entwurf von Vorsteuerungen und Vorfiltern (2-Freiheitsgrade-Struktur)
- Polplatzierung als Entwurfsmethode für Zustandsregler

Lehrmethoden und Lernformen

Tafel, Live Vorführungen der computergestützten Werkzeuge, Texte für das Selbststudium, Praktikumsversuche in simulierten Szenarien

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Regelungstechnik /1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen; Lunze, Jan
- Regelungstechnik /2: Mehrgrößensysteme, digitale Regelung; Lunze, Jan
- Grundlagen Regelungstechnik : einfache Übungen, praktische Beispiele und komplexe Aufgaben; Heinrich, Berthold

711 - Bachelorarbeit mit Bachelorseminar

Empfohlenes Studiensemester: 7
Turnus: fortlaufend
Sprachen: Deutsch, Englisch
ECTS: 12 (BA) + 3 (S) = 15
Prüfungsform: BA (0,8) und Präs (0,2), weitere Angaben siehe Studienplan
Lehrform und SWS: S (1SWS)
Gesamter Workload: 360 Stunden (BA) + 90 Stunden (S) = 450 Stunden
Präsenzzeit: 15 Stunden Seminar
Selbststudium: 435 Stunden
Modulverantwortung: Prof. Dr. Veronika Thurner (FK07)
Weitere Lehrende: alle prüfberechtigten Lehrenden der Hochschule München

Empfohlene Vorkenntnisse für die Teilnahme

Formale Voraussetzung entsprechend der SPO: absolviertes Praxissemester und der Erwerb von 150 ECTS-Kreditpunkten aus Modulen des Studiengangs.

Lernziele

Fach- und Methodenkompetenz

Die Studierenden sind in der Lage,

- die im Studium erworbenen fachlichen und methodischen Kenntnisse und Kompetenzen zur weitgehend selbständigen Bearbeitung eines größeren, aber zeitlich klar begrenzten, praxisbezogenen Projektes einzusetzen.
- eine Literaturrecherche durchzuführen und Fachinformationsquellen für die Arbeit zu nutzen.
- Experimente oder Systemimplementationen zu planen, durchzuführen und die Ergebnisse zu evaluieren.
- die Ergebnisse ihrer Arbeit schriftlich und mündlich verständlich darzustellen und kritisch mit anderen Fachleuten zu diskutieren.

Inhalt

Selbständige Bearbeitung (unter Anleitung) einer praxisbezogenen Problemstellung auf der Basis wissenschaftlicher und methodischer Ansätze.
Im Bachelorseminar werden die Problemstellungen, Inhalte und Ergebnisse der Bachelorarbeiten präsentiert und diskutiert.

Lehrmethoden und Lernformen

wissenschaftliches Arbeiten

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

Fachliteratur: in Abhängigkeit vom Thema der Bachelorarbeit.

Literatur zur Einführung in das wissenschaftliche Arbeiten und Schreiben:

- George D. Gopen and Judith A. Swan: The Science of Scientific Writing, American Scientist, Nov. 1990, Volume 78, pp. 550-558
- Donald E. Knuth et al., Mathematical Writing, MAA Notes, The Mathematical Association of America, 1989, Number 14
- Bernd Heesen: Wissenschaftliches Arbeiten, 4. Auflage, Springer Gabler, 2021

712 - Robotik

Empfohlenes Studiensemester: 7
Turnus: Wintersemester
Sprachen: Deutsch
ECTS: 5
Prüfungsform: schrP oder ModA, weitere Angaben siehe Studienplan
Lehrform und SWS: SU (2SWS), Pra (2SWS)
Gesamter Workload: 150 Stunden
Präsenzzeit: 30 Stunden Seminaristischer Unterricht, 30 Stunden Praktikum
Selbststudium: 90 Stunden
Modulverantwortung: Prof. Dr. Norbert Nitzsche (FK03)

Empfohlene Vorkenntnisse für die Teilnahme

Kompetenzen, die in den Veranstaltungen

- Regelungstechnik
- Technische Mechanik

erworben werden

Lernziele

Fach- und Methodenkompetenz

Die Studierenden erlernen die Grundlagen der Modellierung von Mehrkörpersystemen, insbesondere für mobile Plattformen und serielle kinematische Ketten. Sie können den Zusammenhang zwischen Arbeits- und Konfigurationsraum mathematisch beschreiben und verstehen in diesem Zusammenhang die Begriffe Holonomie und Singularität. Die Studierenden sind in der Lage, einfache geregelte Bewegungen von mobilen Plattformen und seriellen Kinematiken zu erzeugen.

Inhalt

- Koordinatentransformationen
- Methode von Denavit/Hartenberg
- Aufstellen von Bewegungsdifferentialgleichungen für Mehrkörpersysteme
- Bewegungsregelung auf Grundlage der Differentiellen Kinematik
- lokale Gelenkregler, Entkopplung, Zustandsregler

Lehrmethoden und Lernformen

Tafel, Live Vorführungen der computergestützten Werkzeuge, Texte für das Selbststudium, Praktikumsversuche in simulierten Szenarien

Verwendbarkeit des Moduls

Bachelor Digital Engineering

Literatur

- Regelungstechnik /2: Mehrgrößensysteme, digitale Regelung; Lunze, Jan
- Encyclopedia of Robotics; Hsg.: Marcelo H Ang, Oussama Khatib, Bruno Siciliano