



University of Applied Sciences

**HOCHSCHULE
EMDEN·LEER**

Fachbereich Technik

Modulhandbuch

**Bachelor Informatik
(Prüfungsordnung: Version 2024)**

Inhaltsverzeichnis

Pflichtmodule

| | |
|--------------------------------------|----|
| Arbeitstechniken..... | 4 |
| Einführung in die Informatik..... | 5 |
| Hardwaregrundlagen..... | 7 |
| Mathematik 1..... | 10 |
| Mensch-Computer-Interaktion..... | 11 |
| Programmieren 1..... | 13 |
| Betriebswirtschaftslehre..... | 14 |
| C/C++..... | 16 |
| Mathematik 2..... | 18 |
| Programmieren 2..... | 19 |
| Rechnernetze..... | 21 |
| Algorithmen und Datenstrukturen..... | 23 |
| Datenbanken..... | 24 |
| Grundlagen der IT-Sicherheit..... | 26 |
| Hardwarenahe Programmierung..... | 28 |
| Mathematik 3..... | 30 |
| Theoretische Informatik..... | 31 |
| Betriebssysteme..... | 33 |
| Data Science..... | 34 |
| Internet-Technologien..... | 36 |
| Modellierung..... | 38 |
| Rechnerorganisation..... | 40 |
| Echtzeitdatenverarbeitung..... | 42 |
| Projektgruppe..... | 44 |
| Softwareprojektmanagement..... | 46 |
| Informatik und Gesellschaft..... | 48 |
| Parallele und verteilte Systeme..... | 49 |
| Projektarbeit..... | 51 |
| Software-Qualitätssicherung..... | 52 |
| Praxisphase..... | 54 |
| Bachelorarbeit mit Kolloquium..... | 55 |

Wahlpflichtmodule

| | |
|--|----|
| Maschinelles Lernen 1..... | 56 |
| Maschinelles Lernen 2..... | 57 |
| Maschinelles Sehen..... | 59 |
| Spezielle Themen der Datenwissenschaft..... | 61 |
| Ethical Hacking und Pentesting..... | 62 |
| Kryptologie..... | 64 |
| Softwaresicherheit..... | 66 |
| Spezielle Verfahren der IT-Sicherheit..... | 68 |
| Digitaltechnik..... | 70 |
| Eingebettete Systeme..... | 72 |
| HW/SW Codesign..... | 74 |
| Hardwareentwurf mit VHDL..... | 76 |
| Computergrafik..... | 78 |
| Drahtlose Sensortechnik..... | 79 |
| Englisch..... | 80 |
| Kalkulation und Teamarbeit..... | 81 |
| Kommunikation in Marketing und Vertrieb..... | 83 |

| | |
|--------------------------------------|----|
| Marketing für Ingenieure..... | 85 |
| Mediendramaturgie..... | 87 |
| Mixed Reality..... | 88 |
| Produktion Digitaler Medien..... | 90 |
| Spezielle Themen der Informatik..... | 92 |
| Systemprogrammierung..... | 93 |
| Vertriebsprozesse..... | 94 |
| Visuelle Effekte..... | 96 |
| iOS-Programmierung..... | 98 |
| Gamification..... | 99 |

| Modulname | Nummer |
|------------------|--------|
| Arbeitstechniken | 1010 |

| | |
|------------------------------|---|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 1 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Hausarbeit oder Projektbericht oder Klausur 1,5 h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• Arbeitstechniken (2 SWS): N.N.• Praktikum Arbeitstechniken (2 SWS): N.N. |
| Lehrinhalte |
| Studier- und Arbeitstechniken inkl. Verfassen wissenschaftlicher Texte; Präsentationstechniken und Diskussionsleitung; Grundlagen des Projektmanagements; Kommunikation mit Gesprächs- und Besprechungstechniken - auch als Projektteam. |
| angestrebte Lernergebnisse |
| Die Studierenden erkennen die Anforderungen der Studiensituation und erlernen, wie man diese erfüllen kann. Außerdem erwerben sie kommunikative Qualifikationen für Studium, für die Praxisphase und für das spätere Berufsleben anhand aktueller überschaubarer Projektthemen aus dem Umfeld der Informatik. Zusätzlich üben sie, wie man in Gruppen zusammenarbeitet, und erwerben erste Kenntnisse in der Anwendung von Projektmanagement. |
| Lehr- und Lernmethoden |
| Vorlesung und Praktikum |
| Literatur |
| Karagiannakis, Evangelia: Lernstrategien und Arbeitstechniken für MINT-Studiengänge: Ein Lehr- und Übungsbuch. Stuttgart (UTB), 2022. Olfert, K.: Kompakt-Training Projektmanagement. Ludwigshafen (Kiehl), 2019 (11), Schultz von Thun, F.: Miteinander reden. Reinbek (Rowohlt), 1981. |
| Modulverantwortlicher/Modulverantwortliche |
| Studiendekan |
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|------------------------------|--------|
| Einführung in die Informatik | 1020 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 1 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• Einführung in die Informatik (3 SWS): C. Link, N. Streekmann• Praktikum Einführung in die Informatik (1 SWS): C. Link, N. Streekmann |
| Lehrinhalte |
| Formate zur Darstellung von Zahlen sowie Umwandlung dazwischen; Datenspeicherungsformate und Abfragewerkzeuge; Nutzung textbasierter Nutzungsschnittstellen: Shell, SQL, REPL und git; Programmierung; Automaten und reguläre Ausdrücke; Datenstrukturen Records, Arrays und Listen; Einführung Netzwerkprotokolle IP, DNS, TCP, HTTP; Aufbau von Rechnersystemen und Betriebssystemen. Neben den fachlichen Inhalten werden auch historische Entwicklungen und ethische Fragen beleuchtet. |
| angestrebte Lernergebnisse |
| Die Studierenden werden mit den Grundlagen vertraut gemacht, die im weiteren Verlauf des Studiums benötigt werden. Weiterhin wird ein Überblick verschafft über die Problemstellungen die im späteren Studium der Informatik behandelt werden. Studierende kennen verschiedene digitale Darstellungsarten von Zahlen und anderen Daten und können selbst zwischen diesen umwandeln und Programme zur Umwandlung erstellen und testen. Studierende können einige wichtige Datenstrukturen umsetzen und in Programmen verwenden. Studierende beherrschen die grundlegende Verwendung von textbasierten Nutzungsschnittstellen und verstehen das Zusammenspiel von Hardware, Betriebssystem und Software. Studierende sind in der Lage ein kleines Netzwerk zu konfigurieren und Fehler in sehr einfachen Netzkonfigurationen zu finden. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| Vorlesungsskripte und Praktikumsmaterialien werden zur Verfügung gestellt. |

Modulverantwortlicher/Modulverantwortliche

C. Link

Verwendbarkeit

BI, BIPV

| Modulname | Nummer |
|--------------------|--------|
| Hardwaregrundlagen | 1030 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 1 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 2 h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• Grundlagen der Digitalisierung (2 SWS): D. Rabe• Elektrotechnische Hardwaregrundlagen (1 SWS): M. Masur• Praktikum Elektrotechnische Hardwaregrundlagen (1 SWS): M. Masur |
| Lehrinhalte |
| Wichtige Bauelemente, wie z.B. Widerstände, Dioden und Transistoren werden hinsichtlich ihres Aufbaus, ihrer Funktionsweise und ihrer Anwendung beschrieben. Einfache Netzwerke werden dabei dimensioniert, aufgebaut und bezüglich ihres elektrischen Verhaltens untersucht. Digitale Grundfunktionen und kombinatorische Schaltungen werden anhand von Beispielen beschrieben und ebenfalls getestet. |

| angestrebte Lernergebnisse |
|--|
| <p>Qualifikationsziele im Bereich Grundlagen der Digitalisierung(GdD)/Zahlendarstellung und arithmetischer Operationen: Die Studierenden</p> <ul style="list-style-type: none">• stellen ungerichtete und gerichtete ganzzahlige Zahlen als Potenzsumme zu beliebigen Basen dar.• geben die im Bereich der Digitalisierung besonders relevant Basen 2 (dual), 8 (oktal), 16 (hexadezimal inklusive Ziffern A-F für die Dezimalziffern 10 bis 15) an.• berechnen Darstellungen für gerichtete und ungerichtete Zahlen mit unterschiedlichen Basen und addieren/subtrahieren diese handschriftlich.• unterscheiden die Größeneinheiten des Dezimalpräfixes von den Größeneinheiten des Dualpräfixes nach Si- und IEC60027-2 bzw. IEC 80000-13-Normierung.• berechnen gebrochene Zahlen mit unterschiedlichen Basen und bestimmen die Fließkommazahlendarstellung nach IEEE 754 (Zahlendarstellung im Rechner). |
| <p>Qualifikationsziele im Bereich der GdD/Zeichendarstellung: Die Studierenden</p> <ul style="list-style-type: none">• verdeutlichen, dass zur Darstellung von Texten in Rechnern eine Zeichenkodierung in Form von normierten, stetig wachsenden Zeichensätzen erforderlich ist.• bestimmen Zeichencode Paritätsbits.• geben an, dass länderspezifischen ASCII Codes nach ISO/IEC 8859 (druckbare Zeichen) bzw. ISO/IEC 6429 (Steuerzeichen) erweitert wurden.• geben an, dass es neben Codes fester Länge (ASCII, UCS-2, UCS-4 - UCS=universal character set) die Formate UTF-8 und UTF-16 (UCS Transformations Format) gibt, bei denen die Byte-längeninformationen Bestandteil der Codierung ist.• berechnen die in UTF-8 und UTF-16 dargestellten Unicodes unter umgekehrt.• analysieren und synthetisieren in unterschiedlichen Transferformaten dargestellte Texte. |
| <p>Qualifikationsziele im Bereich der GdD/Schaltnetze: Die Studierenden</p> <ul style="list-style-type: none">• stellen digitale (Boolesche) Schaltfunktionen in Wahrheitstabellen und algebraisch als (kanonische) disjunktive sowie konjunktive Normalformen (DNF und KNF) dar.• geben die elementaren Schaltfunktionen inklusive der Schaltsymbole an und erstellen durch deren Zusammenschaltung Schaltpläne zu Booleschen Funktionen.• formen Boolesche Schaltfunktionen (DNF und KNF) mit Hilfe der Booleschen Algebra um und minimieren damit die Funktionsdarstellung - auch unter Anwendung von Karn-augh-Veitch-Diagrammen (KV-Diagrammen).• beurteilen anhand der Verknüpfungszahl den Aufwand zur schaltungstechnischen Realisierung von Booleschen Funktionen. |
| <p>Qualifikationsziele im Bereich der GdD/Schaltwerke/Speicher: Die Studierenden</p> <ul style="list-style-type: none">• analysieren die (speichernden) Schaltfunktionen von digitalen Schaltplänen mit rückgekoppelten elementaren Gatterstrukturen (ungetaktete und taktpiegelgesteuerte Latches, takt-flankengesteuerte Flipflops).• erklären, wie man Latches und Flipflops ansteuern muss, um digitale Informationen zu speichern.• geben an, dass größere Speichermengen in Mikroprozessorschaltungen durch optimierte Speicher realisiert werden.• benennen/klassifizieren unterschiedliche Speicherverfahren (magnetische Speicherung, optische Speicherung, Halbleiter) und ordnen die wesentliche Eigenschaften zu (z.B. flüchtige/nicht flüchtige Speicherung, Hardwareaufwand, Zugriffszeiten). |
| <p>Qualifikationsziele im Bereich Elektrotechnische Hardwaregrundlagen: Die Studierenden lernen elementare Grundlagen der analogen und digitalen Elektronik kennen. Sie sind in der Lage, sowohl passive als auch aktive Bauelemente anzuwenden und die zugehörige Meßtechnik einzusetzen. Dabei wird auch der Unterschied zwischen Theorie und Praxis an ausgewählten Beispielen erläutert und nachgewiesen. Schaltungsanalyse- und Synthese dienen zum komplexen Verständnis elektronischer Baugruppen.</p> |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |

| |
|--|
| Literatur |
| Beuth, K.: Bauelemente (Elektronik 2), Vogel, 2010 Beuth, K.: Digitaltechnik (Elektronik 4), Vogel, 2010 |
| Modulverantwortlicher/Modulverantwortliche |
| M. Masur |
| Verwendbarkeit |
| Bl, BIPV |

| Modulname | Nummer |
|--------------|--------|
| Mathematik 1 | 1040 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 1 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• Mathematik 1 (3 SWS): G. von Cölln• Übung Mathematik 1 (1 SWS): G. von Cölln |
| Lehrinhalte |
| Themen der Logik, diskreten Mathematik und Algebra werden behandelt und das Wissen in Übungen wiederholt und vertieft. Stichworte zu den Inhalten sind: Aussagen- und Prädikatenlogik, Mengen und Relationen, Abzählbarkeit und Kombinatorik sowie Grundlagen der analytischen Geometrie. |
| angestrebte Lernergebnisse |
| Die Studierenden kennen grundlegende Begriffe und Methoden aus der Logik, diskreten Mathematik und lineare Algebra. |
| Lehr- und Lernmethoden |
| Vorlesung, Übung |
| Literatur |
| Teschl, Teschl: Mathematik für Informatiker Band 1 und 2, Springer, 2013 und 2014 Socher, Mathematik für Informatiker, Hanser, 2022 |
| Modulverantwortlicher/Modulverantwortliche |
| G. von Cölln |
| Verwendbarkeit |
| Bl, BIPV |

| Modulname | Nummer |
|-----------------------------|--------|
| Mensch-Computer-Interaktion | 1050 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 1 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Mensch-Computer-Interaktion (2 SWS): T. Pfeiffer • Praktikum Mensch-Computer-Interaktion (2 SWS): T. Pfeiffer |
| Lehrinhalte |
| <p>Theorie</p> <ul style="list-style-type: none"> • Grundlagen der Kognition in Bezug auf Wahrnehmen, Entscheiden und Handeln (z.B. mentale Modelle, Gestaltprinzipien, Handlungsprozesse) • Grundlagen der Gestaltung für grafische Oberflächen (z.B. UI-Patterns, Gestaltungsprinzipien, Styleguides, Normen) • Einsatzgebiete und Typen von Mensch-Maschine-Schnittstellen (e.g. grafische Oberflächen, Sprachdialogsysteme, Mixed-Reality-Systeme) • Methoden und Vorgehensmodelle zur Konzeption von Mensch-Maschine-Schnittstellen (z.B. Human-Centered-Design, Google Design Sprint) • Evaluationstechniken (z.B. Fragebögen, AB-Tests, Interaktionsstudien) <p>Praktischer Teil</p> <ul style="list-style-type: none"> • Bearbeitung von Projekten zur Gestaltung von Mensch-Maschine-Schnittstellen unter Anwendung der in der Vorlesung vermittelten Konzepte und Methoden. <p>Die grundlegenden Lehrinhalte werden in Vorlesungsform vermittelt und praktische Aufgaben in Projektform während der Praktika umgesetzt.</p> |
| angestrebte Lernergebnisse |
| Die Studierenden verstehen die Grundlagen der Wahrnehmung, die Gestaltgesetze und die entsprechenden Modelle der Usability und User Experience. Sie können Softwareoberflächen prozessorientiert gestalten. Sie verwenden hierzu anerkannte Verfahren des Human Centered Design und kennen einschlägige Richtlinien und Normen. Sie kennen gängigste Interaktionsformen und Regeln zum Interaktionsdesign. Im Rahmen des Usability-Engineering können Sie ausgewählte Usability-Methoden exemplarisch anwenden. |

| |
|--|
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| Richter, M.; Flückiger, M.D.: Usability und UX kompakt: Produkte für Menschen, Springer Verlag; 4. Auflage; 2016 |
| Sarodnick, F.; Brau, H.: Methoden der Usability Evaluation, 2. Aufl. Verlag Huber, 2011 |
| Butz, A.; Krüger, A.: Mensch-Maschine-Interaktion, Verlag De Gruyter Oldenbourg; 2014 |
| Modulverantwortlicher/Modulverantwortliche |
| T. Pfeiffer |
| Verwendbarkeit |
| Bl |

| Modulname | Nummer |
|-----------------|--------|
| Programmieren 1 | 1060 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 1 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Programmieren 1 (2 SWS): F. Rump • Praktikum Programmieren 1 (2 SWS): F. Rump |
| Lehrinhalte |
| Elemente der Programmiersprache Java: Literale, Variablen, Datentypen, Ausdrücke und Operatoren, Kontrollstrukturen, Rekursion, Parameterübergabe, Rückgabewerte. Objektorientierte Programmierung: Klassen und Objekte, Methoden, Konstruktoren; Vererbung, Polymorphismus; Ausnahmebehandlung; Ausgewählte Klassen; Dokumentation und Layout von Java-Programmen (JavaDoc). |
| angestrebte Lernergebnisse |
| Die Studierenden verstehen die Grundbegriffe der objektorientierten Programmierung und können eigene einfache Java-Programme erstellen und erläutern. Sie können sich einfache fremde Programme erarbeiten und verstehen. Sie kennen die wichtigsten Programmierrichtlinien und wenden sie in eigenen Programmen an. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| <ul style="list-style-type: none"> • Ratz, D. et al.: Grundkurs Programmieren in Java. Hanser, 2024. • Ullenboom, C.: Java ist auch eine Insel – Einführung, Ausbildung, Praxis. Rheinwerk Computing, 2023. • Schiedermeyer, R.: Programmieren mit Java. Pearson Education, 2004. • Krüger, G., Stark, T.: Handbuch der Java-Programmierung, Addison-Wesley, 2009 |
| Modulverantwortlicher/Modulverantwortliche |
| F. Rump |
| Verwendbarkeit |
| Bl |

| Modulname | Nummer |
|--------------------------|--------|
| Betriebswirtschaftslehre | 1070 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 2 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung oder Kursarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| Betriebswirtschaftslehre (4 SWS): L. Jäncchen |
| Lehrinhalte |
| <ol style="list-style-type: none"> 1. Einordnung und Entwicklung der BWL 2. Ziele, Kennzahlen und Betriebstypen 3. Betriebliche Entscheidungen 4. Konstitutive Entscheidungen 5. Finanz- und Rechnungswesen 6. Betriebliche Leistungserstellung |
| angestrebte Lernergebnisse |
| <p>Die Studierenden kennen und verstehen die wichtigsten Entscheidungsbereiche wirtschaftlichen Handelns in Unternehmen indem Sie:</p> <ul style="list-style-type: none"> • grundlegende Entscheidungen im Rahmen einer Unternehmensgründung beschreiben und mögliche Unternehmensrechtsformen kennen • Grundfunktionen und zugehörige Teilbereiche eines Unternehmens kennen • Aufgaben der Unternehmensführung, wie die Konzeption einer Unternehmensstrategie kennen • Aufgaben der Teilbereiche Beschaffung, Produktion, Absatz und Finanzierung verstehen und wesentliche Zusammenhänge aufzeigen • Investitionsentscheidungen auf der Basis von Investitionsrechnungen treffen • Die Inhalte wesentlicher betriebswirtschaftlicher Begriffe, insbesondere aus dem Rechnungswesen kennen und deren Bedeutung und Zusammenhänge erklären. <p>Dies ermöglicht den Studierenden Ihre technischen Projekte auch im betriebswirtschaftlichen Kontext zu betrachten und so in Ihrem Berufsleben wirtschaftliche Konzepte im Unternehmenskontext anzuwenden.</p> |
| Lehr- und Lernmethoden |
| Vorlesung |

Literatur

Straub, Thomas: Einführung in die Allgemeine Betriebswirtschaftslehre, Pearson Deutschland GmbH, Hallbergmoos, 2. Auflage 2015

Eckardt, Gordon: Business Management - Angewandte Unternehmensführung, GHS, Göttingen, 3. Auflage 2011

Kocian-Dirr, Claudia: Betriebswirtschaftslehre - Schnell erfasst, Springer Gabler, Wiesbaden 2019,

Modulverantwortlicher/Modulverantwortliche

L. Jänchen

Verwendbarkeit

BI, BET, BETPV

| Modulname | Nummer |
|-----------|--------|
| C/C++ | 1080 |

| | |
|------------------------------|---|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 2 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder Erstellung und Dokumentation von Rechnerprogrammen |

| |
|---|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Programmieren 1, Einführung in die Informatik, Mathematik 1 |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • C/C++ (2 SWS): C. Link • Praktikum C/C++ (2 SWS): C. Link |
| Lehrinhalte |
| Zunächst werden die Entwicklungsumgebung auf der Kommandozeile und das Zusammenspiel der einzelnen Werkzeuge besprochen. Die wesentlichen eingebauten Datentypen werden behandelt; danach über benutzerdefinierte Datentypen hin zu Klassen und Objekten und den wichtigsten Typen der Standardbibliothek. Weitere Themen sind Templates sowie Idiome und Design Patterns. |
| angestrebte Lernergebnisse |
| Die studierenden kennen die wesentlichen C/C++-Datentypen und -Abstraktionsmechanismen und können diese sicher auswählen und einsetzen. Sie können bei vorgegebenem Quellcode die einzelnen Vorgänge zur Übersetzungszeit (Präprozessor, Compiler, Linker, etc) und zur Laufzeit (Compiler-generiert) erläutern, in Zusammenhang bringen und gezielt nutzen. Die studierenden sind in der Lage, gängige Programmervorgaben (Style Guides, Best Practices) anzuwenden und darüber hinaus deren Verwendung in fremdem Quelltext kritisch zu betrachten. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| <p>Stroustrup, B: Programming -- Principles and Practice using C++, Addison Wesley Stroustrup, B: A Tour of C++, Addison Wesley Stroustrup, B: The C++ Programming Language, Addison Wesley</p> |
| Modulverantwortlicher/Modulverantwortliche |
| C. Link |

| |
|----------------|
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|--------------|--------|
| Mathematik 2 | 1090 |

| | |
|------------------------------|---|
| ECTS | 10 |
| Semesterwochenstunden | 8 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 2 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 120 h Kontaktzeit + 180 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Mathematik 1 |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Mathematik 2 (6 SWS): G. von Cölln • Übung Mathematik 2 (2 SWS): G. von Cölln |
| Lehrinhalte |
| Weiterführende Themen der diskreten Mathematik, linearen Algebra und Analysis werden behandelt und das Wissen in Übungen wiederholt und vertieft. Stichworte zu den Inhalten sind: Teilbarkeit und modulare Arithmetik, algebraische Strukturen, Graphen, analytische Geometrie, lineare und affine Abbildungen, Matrizen, Gauß-Algorithmus, fehlerkorrigierende Codes, Differential und Integralrechnung mit ein oder mehreren Variablen. |
| angestrebte Lernergebnisse |
| Die Studierenden kennen weiterführende Begriffe und Methoden aus der diskreten Mathematik, linearen Algebra und Analysis und können diese auf konkrete Fragestellungen übertragen. |
| Lehr- und Lernmethoden |
| Vorlesung, Übung |
| Literatur |
| Teschl, Teschl: Mathematik für Informatiker Band 1 und 2, Springer, 2013 und 2014 Socher, Mathematik für Informatiker, Hanser, 2022 |
| Modulverantwortlicher/Modulverantwortliche |
| G. von Cölln |
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|-----------------|--------|
| Programmieren 2 | 1100 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 2 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Programmieren 1 |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Programmieren 2 (2 SWS): F. Rump • Praktikum Programmieren 2 (2 SWS): F. Rump |
| Lehrinhalte |
| Auf Basis der in "Programmieren 1" gelegten Grundlagen werden weitergehende Konzepte der objektorientierten Programmierung vorgestellt und die Verwendung objektorientierter Bibliotheken vertieft. Behandelt werden u.a. Rekursion, Arbeiten mit dem Collections-Framework, Verwendung generischer Datentypen, Datenströme und Dateizugriff, nebenläufige Programmierung mit Threads, Synchronisationsmöglichkeiten bei Zugriff auf gemeinsame Objekte, Netzwerkprogrammierung, Aufbau von Client/Server-Anwendungen, graphische Benutzeroberflächen mit vorgegebenen Komponenten und Ereignisverarbeitung, Verwendung von Lambda-Ausdrücken und Streams. Typische Programmstrukturen werden anhand gängiger Entwurfs- und Architekturmuster (z.B. Model-View-Controller) erläutert. Zur Veranschaulichung werden zu einzelnen Kapiteln praxisnahe Beispiele in Übungsform präsentiert. |
| angestrebte Lernergebnisse |
| Die Studierenden sollen eine konkrete Problemstellung analysieren und algorithmisch lösen können. Sie kennen wichtige Java-Bibliotheken und können diese für konkrete Aufgabenstellungen anwenden. Die Programme werden auf Basis aktueller Werkzeuge erstellt und getestet. Die Studierenden verstehen das Verfahren der testgetriebenen Entwicklung und können dieses für kleine Beispiele anwenden. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| <ul style="list-style-type: none"> • Ratz, D. et al.: Grundkurs Programmieren in Java. Hanser, 2024. • Inden, M.: Der Weg zum Java-Profi. dpunkt.verlag, 2020. • Schiedermeyer, R.: Programmieren mit Java. Pearson Studium, 2010. • Ullensboom, C.: Java ist auch eine Insel -- Einführung, Ausbildung, Praxis. Rheinwerk Computing, 2023. |

Modulverantwortlicher/Modulverantwortliche

F. Rump

Verwendbarkeit

BI

| Modulname | Nummer |
|--------------|--------|
| Rechnernetze | 1110 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 2 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Rechnernetze (3 SWS): D. Kutscher • Praktikum Rechnernetze (1 SWS): D. Kutscher |
| Lehrinhalte |
| Die Grundlagen aus dem Bereich Rechnernetze werden vermittelt: Schichtenmodelle (TCP/IP und OSI) und die Aufgaben sowie die allgemeine Funktionsweise von Diensten und Netzprotokollen. Die Architektur des Internet und die Funktionsweise und Einsatzmöglichkeiten relevanter Netzfunktionen werden ausführlich behandelt. Anhand der TCP/IP-Protokollfamilie werden die Wegewahl und Weiterleitung von Paketen sowie die Transportprotokolle TCP und UDP vertiefend behandelt. Darüber hinaus werden wesentliche Fragestellungen der Netzsicherheit und des Netzmanagements erläutert. Spezielle Netztechnologien wie z.#B. Multicast-Routing, QUIC, VLAN und Funknetze werden anhand von Beispielen betrachtet. |
| angestrebte Lernergebnisse |
| Die Studierenden verstehen grundsätzliche Probleme der Datenkommunikation im Internet und lernen, alternative Lösungsansätze moderner Netzinfrastrukturen (Hardware und Software) zu differenzieren. Die theoretische Grundlage dafür bilden die Eigenschaften und Funktionen des Internet mit einem Schwerpunkt auf den Schichten 2 bis 4 des OSI-Schichtenmodells, damit die Studierenden anschließend in der Lage sind, einfache Kommunikationsnetze nach Vorgabe zu konfigurieren, auf Fehlerfreiheit zu prüfen und anhand vorgegebener Leistungskriterien zu evaluieren. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| Kurose, James; Ross, Keith: Computernetzwerke, 6. Auflage, Pearson, 2014 Tanenbaum, Andrew S.; Feamster, Nick; Wetherall, J.: Computer Networks, 6. Auflage, Pearson, 2021. |

| |
|--|
| Modulverantwortlicher/Modulverantwortliche |
|--|

| |
|-------------|
| D. Kutscher |
|-------------|

| |
|----------------|
| Verwendbarkeit |
|----------------|

| |
|----------------------|
| BI, BET, BETPV, BIPV |
|----------------------|

| Modulname | Nummer |
|---------------------------------|--------|
| Algorithmen und Datenstrukturen | 1120 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 3 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Programmieren 1 |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Algorithmen und Datenstrukturen (2 SWS): N. Streekmann • Praktikum Algorithmen und Datenstrukturen (2 SWS): N. Streekmann |
| Lehrinhalte |
| Es werden häufig verwendete Algorithmen (z.B. Suchverfahren, Sortierverfahren, Wegesuche in Graphen, ...) mit ihren dazu gehörigen Datenstrukturen (z.B. Listen, Bäume, Graphen, ...) vorgestellt und verschiedene Implementierungen bewertet. Es wird besonderer Wert auf die Wiederverwendbarkeit der Implementierungen für unterschiedliche Grunddatentypen gelegt. |
| angestrebte Lernergebnisse |
| Die Studierenden kennen häufig verwendete Algorithmen mit den dazu gehörigen Datenstrukturen und können sie an Beispielen per Hand veranschaulichen. Sie kennen die Laufzeit und den Speicherbedarf der verschiedenen Algorithmen und können einfache Aufwandsanalysen selbstständig durchführen. Sie sind in der Lage zu einer gegebenen Aufgabenstellung verschiedene Algorithmen effizient zu kombinieren und anschließend zu implementieren. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| <p>Sedgewick, R.; Wayne, K.: Algorithms, 4th edition, Addison-Wesley, 2011. Güting, R. H.; Dieker, S.: Datenstrukturen und Algorithmen, 4. Auflage, Springer Vieweg, 2018. Knebl, H.: Algorithmen und Datenstrukturen, 2. Auflage, Springer Vieweg, 2021. Nebel, M.; Wild, S.: Entwurf und Analyse von Algorithmen, 2. Auflage, Springer Vieweg, 2018.</p> |
| Modulverantwortlicher/Modulverantwortliche |
| N. Streekmann |
| Verwendbarkeit |
| BI, BET, BETPV |

| Modulname | Nummer |
|-------------|--------|
| Datenbanken | 1130 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 3 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Programmieren 1 |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• Datenbanken (2 SWS): F. Rump• Praktikum Datenbanken (2 SWS): F. Rump |
| Lehrinhalte |
| <ul style="list-style-type: none">• Begriffe, Konzepte und Architekturen• Relationale Datenbankmanagementsysteme und deren Fundierung• Datenmodellierung (z.B. ER-Modellierung)• Überführung der Modellierung auf ein konkretes Datenmodell (z.B. von ER zu relational)• Normalisierung, Normalformen, Redundanz, Effizienzaspekte• Einführung in eine Anfragesprache (insb. SQL) nebst programmiersprachlichen Erweiterungen• Nutzung von Datenbanken aus Programmiersprachen• Transaktionen und Mehrbenutzerbetrieb• Einführung in fortgeschrittene Datenbanktechnologien |
| angestrebte Lernergebnisse |
| Die Studierenden kennen die grundlegenden Datenbankkonzepte. Sie können komplexe strukturierte Datenumgebungen modellieren und beherrschen deren Abbildung auf relationale Datenbanksysteme. Sie verfügen über vertiefte praktische Kenntnisse im Umgang mit SQL. Die Studierenden sind in der Lage, moderne und etablierte Datenbanktechnologien als Teil komplexer informationstechnischer Projekte einzusetzen. Sie können selbstständig neue Datenbanktechnologien und -konzepte erlernen und in praktische Projekte einfließen lassen. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |

Literatur

- Kleuker, S.: Grundkurs Datenbankentwicklung -- Von der Anforderungsanalyse zur komplexen Datenbankanfrage; 5. Auflage; Springer Vieweg; 2024.
- Adams, R.: SQL Eine Einführung mit vertiefenden Exkursen, Hanser Verlag, 2012.
- Edlich, S. et al.: NoSQL Einstieg in die Welt nichtrelationaler Web 2.0 Datenbanken, 2. Auflage, Hanser, 2011.
- Heuer, A., Saake, G.: Datenbanken - Konzepte und Sprachen, 3. Auflage, mitp, 2008.

Modulverantwortlicher/Modulverantwortliche

F. Rump

Verwendbarkeit

BI

| Modulname | Nummer |
|------------------------------|--------|
| Grundlagen der IT-Sicherheit | 1140 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 3 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Grundlagen der IT-Sicherheit (2 SWS): P. Felke • Praktikum Grundlagen der IT-Sicherheit (2 SWS): P. Felke |
| Lehrinhalte |
| Es werden grundlegende Sicherheitskonzepte und Angriffe vorgestellt und Grundbegriffe der IT-Sicherheit wie Schutzziele, Angreifer, Bedrohungen behandelt. IT-Sicherheitsmechanismen und -standards werden analysiert. Mathematische Grundlagen und erste Techniken aus dem Bereich der Kryptologie sowie des Penetrationtestings werden behandelt. Ganzheitliche Ansätze zur Absicherung von IT-Infrastrukturen auf Basis des IT-Grundschutzes bzw. ISO27001 werden vertieft. Im Praktikum werden diese Themen durch praxisnahe, praktische Übungen vertieft. |
| angestrebte Lernergebnisse |
| Die Studierenden können die wichtigsten Schutzziele, Angreifertypen, Bedrohungen benennen und voneinander abgrenzen. Sie können grundlegende Angriffe und Sicherheitsmaßnahmen beschreiben und Bedrohungen und Risiken für IT-Infrastrukturen beurteilen, implementieren und deren Relevanz einordnen. Die Studierende kennen typische Angriffe, wie XSS oder SQLi und gängige kryptographische Verfahren, wie sie auch in Ransomware genutzt werden. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| IT-Sicherheit: Konzepte - Verfahren - Protokolle, C. Eckert, De Gruyter, Oldenburg Network Hacking, P. Kraft, A. Weyert, FRANZIS 2017 Kryptografie verständlich, Paar, C., Pelzl, J., Springer 2016 |
| Modulverantwortlicher/Modulverantwortliche |
| P. Felke |

Verwendbarkeit

BI, BIPV

| Modulname | Nummer |
|-----------------------------|--------|
| Hardwarenahe Programmierung | 1150 |

| | |
|------------------------------|---|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 3 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder Test am Rechner oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Einführung in die Informatik, C/C++ |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Hardwarenahe Programmierung (2 SWS): C. Koch • Praktikum Hardwarenahe Programmierung (2 SWS): C. Koch |
| Lehrinhalte |
| <p>Das Modul zielt auf die Vermittlung folgender Lehrinhalte: Die generelle Architektur eines Mikroprozessors und sein Zusammenwirken mit dem Speicher, der Rechnerperipherie und einem Betriebssystem. Die Architektur einer Assemblersprache im Vergleich mit höheren Programmiersprachen als auch die eingehende Besprechung des Befehlssatzes der ausgewählten Assemblersprache (i8086-Architektur). Weitere Stichworte sind: Speicherverwaltung, Unterprogrammtechnik und Interruptsystem als Basis des Programmierens in allen höheren Programmiersprachen.</p> |
| angestrebte Lernergebnisse |
| <p>Die Studierenden sollen das Zusammenwirken von Software mit der Hardware eines Rechners verstehen und können sowohl die Struktur einer Assemblersprache als auch ihre wesentlichen Fähigkeiten und die Aufgaben eines Betriebssystems ableiten. Sie kennen hardwarespezifische Grundkonzepte und nutzen diese als Voraussetzung für effizientes Programmieren in höheren Programmiersprachen.</p> |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| <p>Backer, R.: Programmiersprache Assembler, Rowohlt Hamburg, 2007 Erlenkötter, H.: C: Programmieren von Anfang an, Rowohlt Hamburg, 1999 Patterson, D.A.: Rechnerorganisation und -entwurf, Elsevier München, 2005</p> |
| Modulverantwortlicher/Modulverantwortliche |
| C. Koch |

| |
|----------------------|
| Verwendbarkeit |
| BI, BET, BETPV, BIPV |

| Modulname | Nummer |
|--------------|--------|
| Mathematik 3 | 1160 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 3 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Mathematik 1, Mathematik 2 |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• Mathematik 3 (3 SWS): J. Fahlke• Übung Mathematik 3 (1 SWS): J. Fahlke |
| Lehrinhalte |
| Stochastik: Deskriptive Methoden, Kombinatorik, Wahrscheinlichkeitsrechnung, Verteilungen, Tests; |
| angestrebte Lernergebnisse |
| Die Studierenden kennen wichtige Begriffe, Methoden und Verfahren aus der Stochastik. Sie können diese Methoden eigenständig auf anwendungsorientierte Fragestellungen übertragen und die Ergebnisse einordnen und bewerten. |
| Lehr- und Lernmethoden |
| Vorlesung, Übung |
| Literatur |
| Papula, L.: Mathematik für Ingenieure und Naturwissenschaftler Band 3, 7. Auflage, Vieweg +Teubner, 2016. Sachs, M.: Wahrscheinlichkeitsrechnung und Statistik für Ingenieurstudenten an Fachhochschulen, 6. Auflage, Fachbuchverlag Leipzig im Carl Hanser Verlag, 2021. |
| Modulverantwortlicher/Modulverantwortliche |
| J. Fahlke |
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|-------------------------|--------|
| Theoretische Informatik | 1170 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 3 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Programmieren 1, Mathematik 1 |
| Empfohlene Voraussetzung |
| Mathematik 2 |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• Theoretische Informatik (2 SWS): J. Mäkiö• Übung Theoretische Informatik (1 SWS): J. Mäkiö• Praktikum Theoretische Informatik (1 SWS): J. Mäkiö |
| Lehrinhalte |

Stichworte sind: Endliche Automaten, Kellerautomaten, reguläre Ausdrücke, Automaten Transformationen und Minimierung, reguläre und nicht-reguläre Sprachen, Chomsky-Hierarchie, Grammatiken und kontextfreie Sprachen, Berechenbarkeitsmodelle, Churchsche These, Unentscheidbarkeit und Turing-Reduzierbarkeit, Komplexitätsklassen, das P=NP-Problem, polynomiale Reduzierbarkeit, NP-Vollständigkeit.

angestrebte Lernergebnisse

Nach Abschluss dieses Moduls sollen die Studierenden in der Lage sein:

- zu erklären, warum Eingabeinstanzen kodiert werden müssen und was dabei zu beachten ist,
- die wichtigsten mathematischen Grundbegriffe zu erläutern, darunter Mengen, Tupel, Relationen, Funktionen und Graphen,
- die Modelle DEA, NEA und reguläre Ausdrücke sicher zu erklären und ihren Zusammenhang zu erläutern,
- einen NEA, DEA oder regulären Ausdruck für eine reguläre Sprache zu konstruieren,
- die Sprache eines DEA, NEA oder regulären Ausdrucks zu bestimmen,
- einen NEA in einen DEA zu transformieren und den zugehörigen Algorithmus zu erläutern,
- einen DEA zu minimieren und den entsprechenden Algorithmus zu erklären,
- das Pumping-Lemma und dessen Beweisidee zu beschreiben,
- das Pumping-Lemma anzuwenden,
- Kellerautomaten zu erläutern und für eine kontextfreie Grammatik zu konstruieren,
- die Sprache einer kontextfreien Grammatik zu beschreiben,
- die Äquivalenz von kontextfreien Grammatiken und Kellerautomaten zu erläutern,
- die Chomsky-Hierarchie der Sprachen zu erkennen und anhand von Beispielen zu erklären,
- die Chomsky-Normalform zu erläutern und das Verfahren anzugeben sowie anzuwenden,
- das Pumping-Lemma für kontextfreie Grammatiken zu beschreiben und anzuwenden,
- die Abschlusseigenschaften von kontextfreien Grammatiken nennen und anwenden,
- den Zusammenhang zwischen Turing-Maschinen und Berechenbarkeit zu erläutern,
- den Unterschied zwischen "(un)entscheidbarer", "erkennbarer" und "aufzählbarer" Sprache zu erklären,
- die elementaren Komplexitätsklassen zu erläutern und anhand von Beispielen zu erklären.

Das Modul vermittelt grundlegende Kenntnisse im Bereich der theoretischen Informatik. Die Studierenden lernen die Grundbegriffe, Konzepte und Methoden von endlichen Automaten, Grammatiken, Komplexität und Berechenbarkeit kennen sowie die Beziehung zwischen theoretischen Maschinenmodellen und realen Computern.

Lehr- und Lernmethoden

Vorlesung, Praktikum, Studentische Arbeit, Seminar

Literatur

Hopcroft, J.E., Motwani, R., Ullman, J.D.: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie Hedtstück, U.: Einführung in die Theoretische Informatik, Oldenburger Wissenschaftsverlag, 2007. Hoffmann, D.: Theoretische Informatik, Hanser Verlag, 2015.

Modulverantwortlicher/Modulverantwortliche

J. Mäkiö

Verwendbarkeit

BI, BIPV

| Modulname | Nummer |
|-----------------|--------|
| Betriebssysteme | 1180 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 4 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Programmieren 1, C/C++ |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• Betriebssysteme (2 SWS): C. Link• Praktikum Betriebssysteme (2 SWS): C. Link |
| Lehrinhalte |
| Folgende Themen werden behandelt: Dateien, Dateisysteme, Text-basierte Nutzungsschnittstellen und deren Automatisierung, Prozesse, Ausschluss und Synchronisation von Prozessen, Scheduling, Speicherverwaltung. Die Themen werden jeweils aus verschiedenen Blickwinkeln heraus betrachtet (Nutzer, Entwickler, Administrator, Kernel). |
| angestrebte Lernergebnisse |
| Die Entwicklung der Betriebssysteme zeigt, dass sehr viele Konzepte der Informatik für Betriebssysteme entwickelt wurden, die auch in anderen Bereichen der Informatik ihre Anwendung finden. Die Studierenden kennen Methoden, Konzepte, Herausforderungen und Lösungen aus diesem Bereich, so dass sie diese auf ihre zukünftigen Problemstellungen anwenden können. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| Stallings, W.: Operating Systems: Internals and Design Principles, Prentice Hall Silberschatz, A.: Operating System Concepts, Wiley Tanenbaum, A.: Moderne Betriebssysteme, Pearson |
| Modulverantwortlicher/Modulverantwortliche |
| C. Link |
| Verwendbarkeit |
| Bl, BIPV |

| Modulname | Nummer |
|--------------|--------|
| Data Science | 1190 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 4 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Programmieren 1, Programmieren 2, Mathematik 1 |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Data Science (2 SWS): E. Wings • Praktikum Data Science (2 SWS): E. Wings |
| Lehrinhalte |
| <p>Vorgestellt werden grundlegende Konzepte und Methoden aus den Data Science Bereichen Maschine Learning und Big Data. Stichworte sind:</p> <p>Bereich ML:</p> <ul style="list-style-type: none"> • Grundlagen wie supervised/unsupervised learning * Algorithmen: clustering (hierarchical, top-down vs. bottom-up, k-means), classification, Decision Trees, Random Forest, Apriori, * Neuronale Netze: Perzeptron, Forward/Backward Algorithmus, Deep Learning, Convolutions, Pooling, Einführung Architekturen & Methoden * Evaluation measures: confusion matrix, ROC, Silhouette, unbalanced classes, challenges & pitfalls. <p>Bereich Big Data:</p> <ul style="list-style-type: none"> • Big Data Collection: cleaning & integration, data platforms & the cloud * Big Data Storage: Hadoop, modern databases, distributed computing platforms, MapReduce, Spark, NoSQL/ NewSQL * Big Data Systems: Security, Scalability, Visualisation & User Interfaces * Big Data Analytics: Fast Algorithms, Data Compression, Machine Learning Tools for Big Data Frameworks, Case Studies & Applications (e.g. Medicine, Finance) |
| angestrebte Lernergebnisse |
| Die Studierenden kennen die wesentlichen Konzepte in den Bereichen i) Datenintegration und Datenhaltung ii) Datenanalyse und Wissensmanagement sowie iii) Datenvisualisierung und Informationsbereitstellung. Die Studierenden verstehen die Anforderungen von großen Datensätzen (Big Data), kennen grundlegende Konzepte (z.B. MapReduce) und sind mit aktuellen Big-Data Technologien (z.B. Hadoop, Spark) und grundlegenden Machine Learning (ML) Methoden vertraut und können diese auf praktische Problemstellungen anwenden. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |

Literatur

Freiknecht, Jonas: Big Data in der Praxis: Lösungen mit Hadoop, HBase und Hive. Daten speichern, aufbereiten, visualisieren, Carl Hanser Verlag, 2014

Karau, Holden: Learning Spark: Lightning-Fast Big Data Analysis, O'Reilly, 2015

Ester, Martin: Knowledge Discovery in Databases - Techniken und Anwendungen, Springer Verlag, 2000

Modulverantwortlicher/Modulverantwortliche

Studiendekan

Verwendbarkeit

BI, BIPV

| Modulname | Nummer |
|-----------------------|--------|
| Internet-Technologien | 1200 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 4 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Programmieren 1, Programmieren 2 |
| Empfohlene Voraussetzung |
| Datenbanken |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Internet-Technologien (2 SWS): F. Rump • Praktikum Internet-Technologien (2 SWS): F. Rump |
| Lehrinhalte |
| <p>Die Veranstaltung gibt eine Einführung in wichtige Protokolle, Architekturen und Techniken für moderne Internet-Anwendungen auf Basis der Programmiersprache Java. Neben den Basistechnologien für Internet-Anwendungen (z.B. HTTP, HTML, XML, JSON) werden anhand von Servlets und JSPs die Generierung von Web-Seiten, Lesen und Schreiben von Header-Einträgen, Verarbeitung von Anfrageparametern und Nutzung von Cookies und Sessions zur Zusammenfassung mehrerer Anfragen eines Benutzers erläutert.</p> <p>Anhand eines konkreten MVC-Frameworks (z.B. JavaServer Faces) wird die Implementierung professioneller Internet-Anwendungen dargestellt und dessen Vorteile vermittelt. Detailliert wird auf das Bearbeitungsmodell, die Konvertierung von Datentypen, die Validierung der Benutzereingaben, Internationalisierung (Zahlen- und Datumsformate), die Ereignisverarbeitung, die Navigation und die Verwendung von Templates eingegangen. Zur Erhöhung der Interaktivität einer Internet-Anwendung wird das Konzept von Ajax dargestellt und verwendet. Der Zugriff auf relationale Datenbanken zur Bereitstellung der Daten einer Internet-Anwendung wird anhand der Nutzung von JDBC erläutert.</p> <p>Größere Anwendungsbeispiele demonstrieren dabei die vermittelten Lehrinhalte.</p> |
| angestrebte Lernergebnisse |
| Die Studierenden kennen verschiedene Protokolle, Architekturen und Techniken für moderne Internet-Anwendungen. Sie sind in der Lage, unterschiedliche Möglichkeiten zur Implementation von Internet-Anwendungen einzuschätzen und selbst mit einer Auswahl an Techniken Internet-Anwendungen mit Datenbankanbindung zu entwickeln. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |

Literatur

- Müller-Hofmann, F., Hiller, M., Wanner, G.: Programmierung von verteilten Systemen und Webanwendungen mit Java EE. Springer Vieweg, 2015.
- Müller, B.: JavaServer Faces und Jakarta Server Faces 2.3 - Ein Arbeitsbuch für die Praxis, Hanser, 2021.
- Kurz, M., Marinschek, M.: JavaServer Faces 2.2 -- Grundlagen und erweiterte Konzepte. dpunkt.verlag, 2013.
- Pomaska, G.: Webseiten-Programmierung -- Sprachen, Werkzeuge, Entwicklung. Springer Vieweg, 2012.

Modulverantwortlicher/Modulverantwortliche

F. Rump

Verwendbarkeit

BI, BIPV

| Modulname | Nummer |
|--------------|--------|
| Modellierung | 1210 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 4 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Programmieren 1, Programmieren 2 |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Modellierung (2 SWS): N. Streekmann • Praktikum Modellierung (2 SWS): N. Streekmann |
| Lehrinhalte |
| Vorgehensweisen und Modellierungsansätze für Anforderungen an Softwaresysteme (z.B. UML-Modelle, User Stories, ...). Modellierung technischer Lösungen für die Umsetzung funktionaler und nicht-funktionaler Anforderungen unter Berücksichtigung bewährter Entwurfsprinzipien und Entwurfsmuster. Nutzung von UML für die Entwurfsmodellierung. Umsetzung abstrakter Modelle in Source-Code. Anwendung von Clean-Code-Prinzipien. |
| angestrebte Lernergebnisse |
| <p>Die Studierenden kennen verschiedene Prozessmodelle der Softwareentwicklung mit ihren Phasen und Produkten. Sie können Anforderungen an Softwaresysteme analysieren, die für die Softwareentwicklung relevanten Informationen zu abstrahieren und die Anforderungen auf geeignete Weise modellieren.</p> <p>Die Studierenden kennen die wichtigsten Grundlagen der Softwarearchitektur und sind in der Lage Architekturmodelle zu verstehen und zu analysieren. Sie kennen grundlegende Entwurfsprinzipien und können Sie in Modellen (z.B. UML-Diagramme, Source-Code, ...) umsetzen. Sie können objektorientierte Konzepte wie Vererbung, Polymorphie, ... beim Entwurf von Softwaresystemen und der Umsetzung in Source-Code gezielt einsetzen. Sie verstehen Entwurfsmuster sowie die Konsequenzen ihres Einsatzes und können deren generische Lösungen auf konkrete Problemstellungen übertragen.</p> |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| <p>Pohl, K.; Rupp, C.: Basiswissen Requirements Engineering, 5. Auflage, dpunkt.verlag GmbH, 2021.</p> <p>Balzert, H.: Lehrbuch der Objektmodellierung, 2. Auflage, Spektrum, 2005.</p> <p>Musch, O.: Design Patterns mit Java, Springer Vieweg, 2021.</p> |

Modulverantwortlicher/Modulverantwortliche

N. Streekmann

Verwendbarkeit

BI

| Modulname | Nummer |
|---------------------|--------|
| Rechnerorganisation | 1220 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 4 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Hardwarenahe Programmierung |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Rechnerorganisation (3 SWS): G. von Cölln • Übung Rechnerorganisation (1 SWS): G. von Cölln |
| Lehrinhalte |
| <p>Aufbau und Funktionen von Computern werden vorgestellt. Zu Grunde liegenden Konzepte werden dargestellt und hinsichtlich verschiedener Kriterien bewertet.</p> <p>Stichworte zu den Lehrinhalten: Grundlegende Begriffe, Funktion und Aufbau von Computern, Maßnahmen zur Leistungssteigerung, Speicherhierarchien, virtuelle Speicherverwaltung.</p> <p>Es wird besonderer Wert auf die grundlegenden Konzepte sowie auf die Übertragbarkeit auf andere Problemstellungen gelegt.</p> |
| angestrebte Lernergebnisse |
| <p>Die Studierenden verfügen über ein fundiertes, anwendungsorientiertes Wissen über den prinzipiellen Aufbau und die Arbeitsweise von Computern. Sie kennen die wesentlichen Komponenten und deren Zusammenwirken. Die Studierenden können die Leistungsfähigkeit von Computern beurteilen und sind in der Lage diese zu optimieren. Die Studierenden können die grundlegenden Konzepte moderner Computer in anderen technischen Systemen wieder erkennen bzw. diese zur Lösung eigener Aufgabenstellungen anwenden.</p> |
| Lehr- und Lernmethoden |
| Vorlesung, Übung |
| Literatur |
| <p>Patterson, Hennessy: Rechnerorganisation und Rechnerentwurf: Die Hardware/Software-Schnittstelle (De Gruyter Studium), 2022 Patterson, Hennessy: Computer Organization and Design MIPS Edition (Morgan Kaufmann), 2020</p> |
| Modulverantwortlicher/Modulverantwortliche |
| G. von Cölln |

| |
|----------------------|
| Verwendbarkeit |
| BI, BET, BETPV, BIPV |

| Modulname | Nummer |
|---------------------------|--------|
| Echtzeitdatenverarbeitung | 1230 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Hardwarenahe Programmierung |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Echtzeitdatenverarbeitung (2 SWS): A. W. Colombo • Praktikum Echtzeitdatenverarbeitung (2 SWS): A. W. Colombo |
| Lehrinhalte |
| Folgende Inhalte werden vermittelt: Raum- und Zeitbegriff, Echtzeitbetrieb, Hard-und Soft-Echtzeit, Scheduling, Dispatching, Worst-Case-Execution-Time-Analyse (WCET-Analyse) Architekturen von Echtzeitsystemen mit einem Prozessor oder mehrkernigen Prozessoren (z.B. in einem Industrie 4.0-fähige Infrastruktur). Besonderheiten der Systemhardware, mehrkerniger Prozessoren, Entwurf und Implementierung von verteilten Cyber-physischen Systemen. Verifikation, Schedulability, Determinismus, Redundanz, Zuverlässigkeit und Sicherheit, Entwicklungswerzeuge zur Modellierung, Validierung und Konfiguration von verteilten (asynchronous) ereignisorientierten Systemen. Synchronization von nebenläufigen Prozessen. Im Praktikum werden die Kenntnisse mit der Echtzeit-Automatisierung eines komplexen reales Fertigungssystem vertieft (Computer Integrated Manufacturing (CIM) Ebene 1-2). |
| angestrebte Lernergebnisse |
| Die Studierenden werden in der Lage sein, zwei wesentliche Faktoren der Softwareentwicklung von Echtzeitsystemen, "Zeit" und "Hardware", beherrschen zu können. Ihre Kenntnisse über cyber-physische Systeme, Modellierungs- und Analysemöglichkeiten wird sie befähigen Echtzeitapplikationen im Sinne von Model Driven Engineering (MDA) zu realisieren. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |

Literatur

Marwedel, P.: Eingebettete Systeme, Springer 2007
Levi, S.-T., Agrawala, A.K.: Real Time System Design, McGraw-Hill 1990
EU FP7 Project T-CREST - Public Reports 2012-2014
T. Ringler: Entwicklung und Analyse zeitgesteuerter Systeme. at – Automatisierungstechnik/Methoden und Anwendungen der Steuerungs-, Regelungs- und Informationstechnik. 2009
A Survey on Edge and Edge-Cloud Computing Assisted Cyber-Physical Systems, doi: 10.1109/TII.2021.3073066.
DIN SPEC 91345: The Reference Architectural Model Industrie 4.0 (RAMI 4.0). Industrie 4.0 Platform.
Course Skript

Modulverantwortlicher/Modulverantwortliche

A. W. Colombo

Verwendbarkeit

BI, BET, BETPV, BIPV

| Modulname | Nummer |
|---------------|--------|
| Projektgruppe | 1240 |

| | |
|------------------------------|--|
| ECTS | 10 |
| Semesterwochenstunden | 3 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 45 h Kontaktzeit + 255 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Projektbericht |

| |
|---|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Programmieren 1, Programmieren 2, Datenbanken, Mathematik 1, Mathematik 2 |
| Empfohlene Voraussetzung |
| Modellierung, Algorithmen und Datenstrukturen, Data Science |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Projektseminar (2 SWS): Prüfungsbefugte laut BPO-A • Einführung in das wissenschaftliche Arbeiten (1 SWS): Prüfungsbefugte laut BPO-A |
| Lehrinhalte |
| Ausgewähltes Thema aus den Fachthemen des Studiengangs |
| angestrebte Lernergebnisse |
| Die Studierenden sollen die grundlegenden Methoden zur Lösung anspruchsvoller praktischer Probleme in einer Gruppe beherrschen und anwenden können. Hierbei sollen Techniken der Gruppenarbeit, der Kommunikation innerhalb einer Gruppe und der Dokumentation phasenübergreifender Lösungen eingeschätzt und angewendet werden. Die Studierenden können für die Lösung eines ausgewählten und angemessenen forschungs- oder praxisnahen Problems geeignete konzeptionelle oder theoretische Ansätze auswählen, ihre praktische Anwendung auf einen Untersuchungsgegenstand in einer Gruppe organisieren und bewerten, die Implementierung einer Lösung prototypisch durchführen und über diese Ansätze reflektierend mündlich und schriftlich in eigenen Worten berichten. Sie können ein (kleines) Team leiten, die Gruppenarbeit organisieren und Gruppenkonflikte lösen sowie die Auswirkungen des Projektes auf Mitmenschen und Gesellschaft reflektieren. Die Studenten sind in der Lage, eine technische bzw. wissenschaftliche schriftliche Ausarbeitung nach gängigen Methoden zu erstellen. |
| Lehr- und Lernmethoden |
| Seminar, Studentische Arbeit |
| Literatur |
| Literatur themenspezifisch zum gewählten Projekt |
| Modulverantwortlicher/Modulverantwortliche |
| Studiendekan |

Verwendbarkeit

BI

| Modulname | Nummer |
|---------------------------|--------|
| Softwareprojektmanagement | 1250 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Programmieren 1, Programmieren 2, Datenbanken, Algorithmen und Datenstrukturen, Mathematik 1, Mathematik 2 |
| Empfohlene Voraussetzung |
| Data Science, Internet-Technologien, Modellierung |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Softwareprojektmanagement (2 SWS): G. Veltink • Praktikum Softwareprojektmanagement (2 SWS): G. Veltink |
| Lehrinhalte |
| Prozessmodelle der Software-Entwicklung, Rollen und Phasen in den Bereichen: System- bzw. Software-Erstellung, Projektmanagement, Qualitätssicherung und Konfigurationsmanagement. Organisation von Projekten und Funktion des Projektleiters, Projektdefinition, Projektplanung, Projektdurchführung (Projekt-Controlling, Projekt-Kickoff, Vertragsmanagement, Information und Kommunikation), Projektabschluss, Führung von IT-Projekten - auch im Hinblick auf Projektmitarbeiter. |
| angestrebte Lernergebnisse |
| Die Studierenden kennen verschiedene Prozessmodelle. Sie können für überschaubare Aufgabenstellungen die Software-Entwicklung planen, kontrollieren und steuern. Dabei sind sie in der Lage, ihre Entscheidungen zu begründen und gegenüber Auftraggebern zu vermitteln und können mit Konflikten in Gruppen umgehen. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| Hindel, B. u. a.: Basiswissen Software-Projektmanagement. Aus- und Weiterbildung zum certified professional for project management nach ISQI-Standard. Heidelberg, Dpunkt-Verlag, 2009 (3). Olfert, K.: Kompakt-Training Projektmanagement. Ludwigshafen, Kiehl, 2016 (10). Wieczorek, H. W. u. Mertens, P.: Management von IT-Projekten. Von der Planung zur Realisierung. Berlin, Heidelberg, Springer, 2011 (4). |
| Modulverantwortlicher/Modulverantwortliche |
| G. Veltink |

| |
|----------------|
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|-----------------------------|--------|
| Informatik und Gesellschaft | 1260 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder Studienarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| Informatik & Gesellschaft (4 SWS): N. N. |
| Lehrinhalte |
| Juristische Grundlagen: Grundgesetz, BGB und andere Gesetze; IT-Recht; Mediengesetze; Datenschutzgesetze; Urheberrecht; EU-Recht; Fallbeispiele |
| angestrebte Lernergebnisse |
| Die Studierenden kennen die Grundstrukturen und Grundprinzipien des Rechts und des Datenschutzes und können diese auf IT-Fragen übertragen. Sie können Fallbeispiele aus dem IT-Umfeld rechtlich analysieren und Lösungsstrategien für konkrete IT-bezogene Fragestellungen entwickeln und bewerten. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| Ehmann, E.: Datenschutz von A - Z Ausgabe 2016, WEKA Media, 2016. Heise, A., Sodtalbers, A., Volkmann, C.: IT-Recht, W3L, 2010. Taeger, H.: Einführung in das Datenschutzrecht, Fachmedien Recht und Wirtschaft Verlag, 2013. Witt, B. C.: Datenschutz kompakt und verständlich, Vieweg + Teubner, 2010. |
| Modulverantwortlicher/Modulverantwortliche |
| Studiendekan |
| Verwendbarkeit |
| BI |

| Modulname | Nummer |
|---------------------------------|--------|
| Parallele und verteilte Systeme | 1270 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung |

| |
|---|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Echtzeitdatenverarbeitung, Betriebssysteme, Rechnerorganisation, Programmieren 1, Programmieren 2, Theoretische Informatik, C/C++ |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Parallele und verteilte Systeme (3 SWS): G. J. Veltink • Praktikum Parallele und verteilte Systeme (1 SWS): G. J. Veltink |
| Lehrinhalte |
| Konzepte der Parallelverarbeitung auf verschiedenen Ebenen werden vorgestellt und bewertet. Entwicklungsmethoden und Werkzeuge zur nebenläufigen Programmierung, sowie formale Methoden zur Spezifikation von nebenläufigen Prozessen werden vorgestellt und an praktischen Beispielen angewendet. Stichworte sind: Konzepte und Organisationen zur nebenläufigen, parallelen und verteilten Verarbeitung, Interprozesskommunikation, synchrone und asynchrone Kommunikation, entfernte Aufrufe (RPC, RMI), Prozesssalgebra, verteilte Koordination, Einigung und Konsens. Die Veranstaltung gibt eine Einführung in die Theorie nebenläufiger, paralleler und verteilter Systeme sowie deren praktischen Anwendungsbereiche und in die technologischen Grundlagen für die Anwendung verteilter Systeme. |
| angestrebte Lernergebnisse |
| Die Studierenden können die wesentlichen Konzepte der Nebenläufigkeit und der Parallelverarbeitung und deren Realisierung anwenden. Sie können die Einsatzgebiete und Grenzen der Leistungssteigerung durch Parallelverarbeitung analysieren und sie können nebenläufige, parallele und verteilte Programme in Gruppenarbeit erschaffen. Die Studierenden erhalten Kenntnisse über Systeme und Architekturen zur Nutzung paralleler und verteilter Rechnerressourcen und deren Architektur, sowie über die formale Spezifikation und Verifikation von kooperierenden nebenläufigen Prozessen und über grundlegende verteilte Algorithmen. Damit können sie die Vor- und Nachteile von Technologien zur Erstellung paralleler und verteilten Anwendungen analysieren und gegenüberstellen sowie nebenläufige und verteilte Anwendungen formal spezifizieren, analysieren und implementieren, mit dem Ziel im späteren Berufsleben die geeigneten Technologien für praktische Probleme auswählen und einsetzen zu können. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |

Literatur

- Tanenbaum, van Steen: Verteilte Systeme: Prinzipien und Paradigmen, Pearson Studium, 2007.
- Groote, Mousavi: Modeling and Analysis of Communicating Systems, MIT Press, 2014
- McCool et al.: Structured Parallel Programming: Patterns for Efficient Computation, Morgan Kaufmann, 2012

Modulverantwortlicher/Modulverantwortliche

G. J. Veltink

Verwendbarkeit

BI, BIPV

| Modulname | Nummer |
|---------------|--------|
| Projektarbeit | 1280 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 10 h Kontaktzeit + 140 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Projektbericht |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Programmieren 1, Programmieren 2, Datenbanken, Mathematik 1, Mathematik 2 |
| Empfohlene Voraussetzung |
| Modellierung, Algorithmen und Datenstrukturen, Data Science, Softwareprojektmanagement |

| |
|--|
| Lehrveranstaltungen |
| Projektarbeit: Prüfungsbefugte laut BPO-A |
| Lehrinhalte |
| Eine Fragestellung aus der Praxis zu einem oder mehreren Fachgebieten des Studiengangs wird unter realen Bedingungen, bevorzugt in Zusammenarbeit mit einem Industrieunternehmen, bearbeitet. |
| angestrebte Lernergebnisse |
| Die Studierenden erarbeiten eine Lösung einer komplexen, für den Studiengang typischen Fragestellung. Sie kombinieren dabei die in verschiedenen Lehrveranstaltungen separat erlernten Fähigkeiten unter realen Bedingungen. Sie wenden Methoden des Projektmanagements, der Gruppenarbeit und der Kommunikation an und dokumentieren das Projektergebnis. Sie können die Auswirkungen des Projektes auf Mitmenschen und Gesellschaft einschätzen. |
| Lehr- und Lernmethoden |
| Studentische Arbeit |
| Literatur |
| Literatur themenspezifisch zur Projektarbeit |
| Modulverantwortlicher/Modulverantwortliche |
| Studiendekan |
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|-----------------------------|--------|
| Software-Qualitätssicherung | 1290 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung |

| |
|---|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Programmieren 1, Programmieren 2, Modellierung, Softwareprojektmanagement |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Software-Qualitätssicherung (2 SWS): N. Streekmann • Praktikum Software-Qualitätssicherung (2 SWS): N. Streekmann |
| Lehrinhalte |
| Die Rolle von Qualitätssicherung im Softwareentwicklungsprozess, Testen in agilen Projekten, Verfahren zur Erstellung dynamischer Tests in unterschiedlichen Kontexten (z.B. Unit-Test, Integrationstest, Akzeptanz-Test, ...), Test-Driven Development, Mocking, Code Coverage, Statistische Codeanalyse, Review-Verfahren, Automatisierte Testausführung, Umgang mit Fehlern in Anwendungen. |
| angestrebte Lernergebnisse |
| Die Studierenden können die Grundbegriffe der Software-Qualitätssicherung wiedergeben. Sie sind in der Lage Anwendungen auf unterschiedlichen Ebenen systematisch zu testen. Sie kennen unterschiedliche Review-Verfahren und können diese im Hinblick auf die Anwendbarkeit in einem gegebenen Szenario bewerten. Die Studierenden sind in der Lage für gegebene Anwendungstypen und Entwicklungsverfahren passende Qualitätssicherungsmethoden auszuwählen. Sie verstehen die Möglichkeiten der Automatisierung in der Qualitätssicherung und können beispielhaft ausgewählte Werkzeuge in diesem Bereich anwenden. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| Spillner, A.; Linz, T.: Basiswissen Softwaretest: Aus- und Weiterbildung zum Certified Tester. 6. Auflage, dpunkt.verlag, 2019. Baumgartner, M. et al.: Agile Testing: Der agile Weg zur Qualität. 3. Auflage, Hanser, 2023. Linz, T.: Testen in Scrum-Projekten. 3. Auflage, dpunkt.verlag, 2023. |
| Modulverantwortlicher/Modulverantwortliche |
| N. Streekmann |

| |
|----------------|
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|-------------|--------|
| Praxisphase | 1300 |

| | |
|------------------------------|--|
| ECTS | 18 |
| Semesterwochenstunden | 1 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 7 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 15 h Kontaktzeit + 525 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Praxisbericht |

| |
|---|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Programmieren 1, Programmieren 2, Datenbanken, Mathematik 1, Mathematik 2, Modellierung, Algorithmen und Datenstrukturen, Data Science, Softwareprojektmanagement |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Praxisarbeit: Prüfungsbefugte laut BPO-A • Praxisseminar (1 SWS): Prüfungsbefugte laut BPO-A |
| Lehrinhalte |
| Fachthemen entsprechend den Aufgaben im gewählten Betrieb. Alternativ internationale Studien: Bearbeitung von Vorlesungen und Praktika in einer Partnerhochschule. |
| angestrebte Lernergebnisse |
| Ziel der Praxisphase ist es, den Anwendungsbezug der im Studium erworbenen Kenntnisse, Fähigkeiten und Fertigkeiten durch praktische Mitarbeit in einer Praxisstelle (Betrieb) zu erweitern und zu vertiefen. Die Studierenden wissen, welche Anforderungen in der späteren Berufspraxis auf sie zukommen, sind in der Lage, ihre im Studium erworbenen Kenntnisse und Fähigkeiten anzuwenden und die bei der praktischen Tätigkeit gesammelten Ergebnisse und Erfahrungen zu reflektieren und auszuwerten. Sie können selbstständig und überzeugend über das Erarbeitete referieren und schriftlich berichten. Alternativ internationale Studien: Die Studierenden können in einer ausländischen Hochschule in einer fremden Sprache neuen Stoff erarbeiten, sie erkennen die interkulturellen Aspekte. |
| Lehr- und Lernmethoden |
| Studentische Arbeit, Seminar |
| Literatur |
| Literatur themenspezifisch zu den Aufgaben im gewählten Betrieb. |
| Modulverantwortlicher/Modulverantwortliche |
| Studiendekan |
| Verwendbarkeit |
| BI, BET, BETPV, BMT |

| Modulname | Nummer |
|-------------------------------|--------|
| Bachelorarbeit mit Kolloquium | 1310 |

| | |
|------------------------------|--|
| ECTS | 12 |
| Semesterwochenstunden | |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 7 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | PF |
| Arbeitsaufwand | 20 h Kontaktzeit + 340 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Bachelorarbeit mit Kolloquium |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| Bachelorarbeit mit Kolloquium: Prüfungsbefugte laut BPO-A |
| Lehrinhalte |
| Die Bachelorarbeit ist eine theoretische, empirische und/oder experimentelle Abschlussarbeit mit schriftlicher Ausarbeitung, die individuell durchgeführt wird. Die Arbeit wird abschließend im Rahmen eines Kolloquiums präsentiert. |
| angestrebte Lernergebnisse |
| In der Bachelorarbeit zeigen die Studierenden, dass sie in der Lage sind, ein Problem aus den wissenschaftlichen, anwendungsorientierten oder beruflichen Tätigkeitsfeldern dieses Studiengangs selbstständig unter Anwendung wissenschaftlicher Methoden und Erkenntnisse zu bearbeiten und dabei in die fächerübergreifenden Zusammenhänge einzuordnen. Folgende Kompetenzen werden erworben: Kompetenz sich in das Thema einzuarbeiten, es einzuordnen, einzugrenzen, kritisch zu bewerten und weiter zu entwickeln; Kompetenz das Thema anschaulich und formal angemessen in einem bestimmten Umfang schriftlich darzustellen; Kompetenz, die wesentlichen Ergebnisse der Arbeit fachgerecht und anschaulich in einem Vortrag einer vorgegebenen Dauer zu präsentieren; Kompetenz aktiv zu fachlichen Diskussionen beizutragen. |
| Lehr- und Lernmethoden |
| Studentische Arbeit |
| Literatur |
| Literatur themenspezifisch zur Bachelorarbeit |
| Modulverantwortlicher/Modulverantwortliche |
| Studiendekan |
| Verwendbarkeit |
| BI, BET, BETPV, BIPV, BMT |

| Modulname | Nummer |
|-----------------------|--------|
| Maschinelles Lernen 1 | 2010 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung oder Studienarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Mathematik 1, Programmieren 1 |
| Empfohlene Voraussetzung |
| Algorithmen und Datenstrukturen |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Maschinelles Lernen 1 (2 SWS): N. N. • Praktikum Maschinelles Lernen 1 (2 SWS): N. N. |
| Lehrinhalte |
| Die verschiedenen Konzepte von Maschinellem Lernen (überwachtes, unüberwachtes und bestärkendes Lernen) werden vorgestellt und Grundbegriffe der Domäne erläutert. Die Studierenden lernen grundlegende Methoden und Verfahren zur u. A. Regression, Klassifizierung, Clusteranalyse und Entscheidungsfindung mittels praktischer Übungen in Python kennen. |
| angestrebte Lernergebnisse |
| Die Studierenden kennen die verschiedenen Konzepte des Maschinellen Lernens und können einfache Problemstellungen entsprechend einordnen. Sie sind in der Lage, geeignete Verfahren für ein einfaches Problem auszuwählen, anzuwenden und die Ergebnisse zu bewerten. Sie verfügen über vertiefte theoretische und praktische Kenntnisse im Umgang mit einer domänen spezifischen Programmiersprache und Bibliotheken. |
| Lehr- und Lernmethoden |
| Seminar |
| Studiengangsschwerpunkte |
| Wahlpflichtfach Zertifikat Data Science |
| Literatur |
| Russel, S.; Norvig, P.: Artifical Intelligence - A Modern Approach, Pearson, 2021. |
| Modulverantwortlicher/Modulverantwortliche |
| Studiendekan |
| Verwendbarkeit |
| BI, BET, BETPV, BIPV |

| Modulname | Nummer |
|-----------------------|--------|
| Maschinelles Lernen 2 | 2020 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung oder Studienarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Mathematik 1, Mathematik 2, Programmieren 1, Programmieren 2 |
| Empfohlene Voraussetzung |
| Maschinelles Lernen 1, Maschinelles Sehen, Algorithmen und Datenstrukturen, Data Science |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Maschinelles Lernen 2 (2 SWS): N. N. • Praktikum Maschinelles Lernen 2 (2 SWS): N. N. |
| Lehrinhalte |
| Auf Basis des Moduls Maschinelles Lernen 1 lernen die Studierenden weitergehende Konzepte und Methoden (bspw. probabilistische Modelle, Deep Learning) mit praktischen Übungen aus unterschiedlichen Domänen (bspw. Maschinelles Sehen, Computerlinguistik) kennen. Die Studierenden lernen wie Modelle in modulare Systemlandschaften mittels Containerisierung (bspw. Docker, Kubernetes) und Daten-Pipelines (bspw. Apache Kafka, PostgreSQL) integriert und überwacht (bspw. Grafana) werden. |
| angestrebte Lernergebnisse |
| Die Studierenden sind in der Lage konkrete Problemstellungen im Kontext des maschinellen Lernens zu analysieren. Sie kennen wichtige Machine Learning und Deep Learning Bibliotheken und können diese für konkrete Aufgabenstellungen aus unterschiedlichen Domänen anwenden. Die Studierenden verstehen den Prozess der Integration von Modellen in modulare Cloud-Umgebungen und können diesen für einfache Beispiele realisieren. |
| Lehr- und Lernmethoden |
| Seminar |
| Studiengangsschwerpunkte |
| Wahlpflichtfach Zertifikat Data Science |
| Literatur |
| Russel, S.; Norvig, P.: Artifical Intelligence - A Modern Approach, Pearson, 2021. |
| Modulverantwortlicher/Modulverantwortliche |
| Studiendekan |

| |
|----------------|
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|--------------------|--------|
| Maschinelles Sehen | 2030 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung oder Studienarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Mathematik I |
| Empfohlene Voraussetzung |
| Algorithmen und Datenstrukturen, Mathematik 2 |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Maschinelles Sehen (2 SWS): C. Koch • Praktikum Maschinelles Sehen (2 SWS): C. Koch |
| Lehrinhalte |
| <p>Das Modul kombiniert theoretische Grundlagen mit praktischen Übungen und Projekten, um den Studierenden ein umfassendes Verständnis der Bildverarbeitung und des maschinellen Sehens zu vermitteln. Als Software-Werkzeug zur Analyse und Visualisierung von Bild- und Sensordaten dient hierbei Python oder Matlab/Simulink.</p> <p>Stichworte: Anwendungsgebiete und Entwicklung des maschinellen Sehens, Bildsensorik, optische Abbildung, Bildvorverarbeitung durch Signalfilterung, Kontrastverbesserung und Rauschunterdrückung, morphologische Operatoren, Verfahren zur Bildsegmentierung, Merkmalsextraktion, Mustererkennung mittels k-Nearest-Neighbor-Algorithmus, Bayes-Klassifikator und Neuronalen Netzen</p> |
| angestrebte Lernergebnisse |
| <p>Maschinelles Sehen (engl. Machine Vision) ist ein Teilbereich des maschinellen Lernens im Grenzbereich zwischen Informatik und den Ingenieurwissenschaften, aufbauend auf Algorithmen aus der digitalen Bild- und Signalverarbeitung.</p> <p>Das Modul zielt darauf ab, den Studierenden grundlegende Kenntnisse und Fähigkeiten im Bereich der Bildverarbeitung und des maschinellen Sehens zu vermitteln. Die Studierenden sollen in der Lage sein, komplexe visuelle Daten einzuordnen und maschinell analysieren, interpretieren und verarbeiten zu lassen. Sie sollen die Grundlagen moderner Algorithmen und Techniken des maschinellen Sehens verstehen und anwenden können. Darüber hinaus sollen sie in der Lage sein, einfache Bildverarbeitungsaufgaben in verschiedenen Anwendungsbereichen im industriellen Umfeld praktisch zu lösen.</p> |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Studiengangschwerpunkte |
| Wahlpflichtfach Zertifikat Data Science |

Literatur

Gonzalez, R.C. und Woods, R.E.: Digital Image Processing, Prentice Hall, 4rd edition, 2017
Szeliski, R.: Computer Vision: Algorithms and Applications, Springer, 2nd edition 2022
Corke P.: Robotics, Vision and Control, Springer Verlag Berlin, 2013

Modulverantwortlicher/Modulverantwortliche

C. Koch

Verwendbarkeit

BI, BET, BETPV, BIPV

| Modulname | Nummer |
|--|--------|
| Spezielle Themen der Datenwissenschaft | 2040 |

| | |
|------------------------------|---|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung oder Studienarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Mathematik I, Programmieren 1 |
| Empfohlene Voraussetzung |
| Algorithmen und Datenstrukturen, Data Science, Maschinelles Lernen 1 |

| |
|---|
| Lehrveranstaltungen |
| Spezielle Themen der Datenwissenschaft (4 SWS): C. Koch |
| Lehrinhalte |
| Das Modul hat zum Ziel eine anwendungsbezogene, vertiefende Betrachtung der in den Modulen "Maschinelles Lernen 1" und "Maschinelles Sehen" behandelten Themen und Algorithmen. Hierbei sollen vielfältige, aktuelle und wechselnde Aspekte behandelt werden, ausgelegt als seminaristische Lehrveranstaltung mit wechselnden Dozenten aus der Informatik, Elektronik, Maschinenbau und Betriebswirtschaft. Die daraus folgenden aktuellen Lehrinhalte werden den Studierenden vor Semesterbeginn bekanntgegeben. |
| angestrebte Lernergebnisse |
| Die Studenten kennen aktuelle Themen der Datenwissenschaft (Data Science), können sich selbstständig in auftretende Themen und Probleme einarbeiten und Lösungen nach Stand der Technik entwickeln und präsentieren. |
| Lehr- und Lernmethoden |
| Vorlesung oder Seminar |
| Studiengangschwerpunkte |
| Wahlpflichtfach Zertifikat Data Science |
| Literatur |
| Wird den Studierenden vor Beginn der Veranstaltung bekanntgegeben. |
| Modulverantwortlicher/Modulverantwortliche |
| C. Koch |
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|--------------------------------|--------|
| Ethical Hacking und Pentesting | 2050 |

| | |
|------------------------------|---|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung oder Kursarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Grundlagen der IT-Sicherheit, Rechnernetze, Grundlagen der Programmierung oder C/C++ |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Ethical Hacking und Pentesting (2 SWS): P. Felke • Praktikum Ethical Hacking und Pentesting (2 SWS): P. Felke |
| Lehrinhalte |
| Es werden Schwachstellen von IT-Infrastrukturen, mobilen Kommunikationsnetzwerken und Sicherheitsprotokollen vorgestellt, wie z.B. Angriffe gegen das Active Directory, WLAN, TLS, oder mittels Buffer-Overflows, sowie Gegenmaßnahmen behandelt. Hierbei werden insbesondere allgemeine Angriffstechniken an praktischen Beispielen vermittelt, um selbst neue zu entwickeln zu können aber auch Strategien, um IT-Infrastrukturen abzusichern. Die Angriffe und entsprechenden Sicherheitslösungen werden im Praktikum analysiert, bewertet und implementiert. |
| angestrebte Lernergebnisse |
| Die Studierenden kennen Schwachstellen und Angriffsmethoden auf IT-Infrastrukturen, mobile Kommunikationsnetzwerke bzw. Sicherheitsprotokollen. Durch die Analyse und Bewertung der Schwachstellen können Pentests durchgeführt und Gegenmaßnahmen identifiziert werden, die dann unter Anwendung ausgewählter Werkzeuge und unter Berücksichtigung rechtlicher Rahmenbedingungen implementiert werden. Dadurch können die Studierenden später gegeeignete Pentests entwickeln um IT-Infrastrukturen zu untersuchen und die Kritikfähigkeit der entdeckten Schwachstellen bewerten. Sie sind in der Lage Sicherheitslücken zu schließen aber auch Angriffstools (weiter)zuentwickeln. Die Grenze zwischen technischer Machbarkeit und sozialer bzw. ethischer Verantwortung ist den Studierenden bewusst. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum, Studentische Arbeit |
| Studiengangschwerpunkte |
| Wahlpflichtfach Zertifikat IT-Sicherheit |

| |
|--|
| Literatur |
| O'Gorman, K., Kearns, D., Kennedy, D., Aharoni, M.: Metasploit: Die Kunst des Penetration Testing, mitp professional J. Erickson: Hacking: Die Kunst des Exploits, dpunkt.verlag J. Schwenk: Sicherheit und Kryptographie im Internet, Springer 2016 |
| Modulverantwortlicher/Modulverantwortliche |
| P. Felke |
| Verwendbarkeit |
| BI, BET, BETPV, BIPV |

| Modulname | Nummer |
|-------------|--------|
| Kryptologie | 2060 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 4 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung oder Kursarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Programmieren 1 oder C/C++ |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Kryptologie (2 SWS): P. Felke • Übung Kryptologie (2 SWS): P. Felke |
| Lehrinhalte |
| Symmetrische und asymmetrische Kryptographie, wie z.B. AES, RSA und die Familie der SHA-Hashfunktionen, werden vorgestellt. Die mathematischen, algorithmischen und kryptoanalytischen Aspekte werden erläutert. Es werden die Einsatzzwecke, z.B. in Übertragungsprotokollen, erklärt und diskutiert. Die Angriffe bzw. kryptoanalytischen Ansätze, so wie die Verfahren selbst werden im Praktikum vertieft, bewertet und implementiert. |
| angestrebte Lernergebnisse |
| Die Studierenden kennen grundlegende Algorithmen für symmetrische und asymmetrische Verschlüsselung, sowie die wesentlichen Angriffsmethoden. Sie kennen Einsatzsszenarien von asymmetrischer, symmetrischer Kryptographie und sind dadurch in der Lage geeignete Verfahren für bestimmte Anwendungszwecke auszuwählen. Die Studierenden können moderne Kryptosysteme mit den gelernten Angriffstechniken kryptoanalysieren und entsprechende Hilfsprogramme erstellen. Sie sind dadurch in der Lage die Güte von Verfahren hinsichtlich dieser Angriffe zu bewerten. Sie kennen typische, effiziente Algorithmen zur Implementation von Kryptosystemen und Fallstricke bei der Umsetzung bzw. beim Einsatz in Übertragungsprotokollen. Die Studierenden sind dadurch auch in der Lage, die eingesetzten kryptographischen Verfahren in typischen Übertragungsprotokollen, wie z.B. TLS, sicherheitlich zu bewerten. |
| Lehr- und Lernmethoden |
| Vorlesung, Übung, Studentische Arbeit |
| Studiengangschwerpunkte |
| Wahlpflichtfach Zertifikat IT-Sicherheit |

Literatur

Paar, C., Pelzl, J.: Kryptografie verständlich, Springer 2016
Buchmann, J.: Einführung in die Kryptographie, Springer 2010
Stinson, D.: Cryptography, Theory and Practice, fourth Edition, CRC Press 2019

Modulverantwortlicher/Modulverantwortliche

P. Felke

Verwendbarkeit

BI, BIPV

| Modulname | Nummer |
|--------------------|--------|
| Softwaresicherheit | 2070 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Kursarbeit oder Klausur 1,5h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Programmieren 1 oder C/C++ |
| Empfohlene Voraussetzung |
| Betriebssysteme |

| |
|--|
| Lehrveranstaltungen |
| Softwaresicherheit (4 SWS): C. Link |
| Lehrinhalte |
| Schwachstellen wie Pufferüberlauf, Rechteerweiterung, Bedrohungsanalyse, etc. Gegenmaßnahmen wie Ausführungsverhinderung, Codesignaturen, Sandboxes. Erweiterte Sicherheitsmechanismen von Betriebssystemen (SELinux, Windows, BSD-basierte). Sicherheitsarchitekturen von Programmiersprachen und -frameworks (z. B. Java, C#). Sicherheitsregelwerke wie PCI-DSS und Common Criteria. Verschiedene Ausprägungen von Zugriffskontrolle mit dazugehörigen Richtlinien. |
| angestrebte Lernergebnisse |
| Die Studierenden kennen Schutzziele, Bedrohungen, Gegenmaßnahmen und deren Zusammenhang im Softwarestapel Betriebssystem, Compiler, Ablaufumgebung, Bibliothek und Programm. Die Studierenden können so Sicherheitslücken vermeiden und durch das Einbringen (bzw. Aktivieren und Konfigurieren) von Schutzmechanismen die Sicherheit beim Betrieb von Software erhöhen. Sie kennen verschiedene Ausprägungen von Zugriffskontrollen mit dazugehörigen Richtlinien. |
| Lehr- und Lernmethoden |
| Seminar |
| Studiengangsschwerpunkte |
| Wahlpflichtfach Zertifikat IT-Sicherheit |
| Literatur |
| Howard M, Le Blanc, D.: Writing Secure Code, Microsoft Press Books, 2. Auflage 2003 Oaks, S.: Java Security, O Reilly and Associates, 2. Auflage 2001 |
| Modulverantwortlicher/Modulverantwortliche |
| C. Link |

| |
|----------------------|
| Verwendbarkeit |
| BI, BET, BETPV, BIPV |

| Modulname | Nummer |
|---------------------------------------|--------|
| Spezielle Verfahren der IT-Sicherheit | 2080 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung oder Kursarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Grundlagen der IT-Sicherheit |

| |
|---|
| Lehrveranstaltungen |
| Seminar Spezielle Verfahren der IT-Sicherheit (4 SWS): P. Felke |
| Lehrinhalte |
| Aktuelle Themen der IT-Sicherheit oder Forschungsthemen werden vorgestellt und Aufgabenstellungen formuliert, die von den Studenten selbstständig zu bearbeiten und vorzutragen sind. Beispiele für Themenbereiche (stichwortartig): aktuelle Verfahren aus dem Bereich Computer-, Netzwerksicherheit, mobile Endgeräte und Satellitentelefone, sowie deren Angriffsmethoden. Aktuelle Entwicklungen aus dem Bereich Ransomware, Virenentwicklung und -scannern, Cloudcomputing oder eGovernment. Das Lösen einer Cryptochallenge in Gruppenarbeit. Diese basiert auf der Post-Quanten-Kryptographie. |
| angestrebte Lernergebnisse |
| Die Studenten kennen aktuelle Themen der IT-Sicherheit, können sich selbstständig in auftretende Themen und Probleme einarbeiten und Lösungen nach Stand der Technik entwickeln, aufschreiben und präsentieren. Durch die Cryptochallenge lernen die Studierenden wie in der Praxis Crackingsoftware entwickelt wird am Beispiel der Post-Quanten-Kryptographie. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum oder Seminar (Seminar mit Anwesenheitspflicht) |
| Studiengangsschwerpunkte |
| Wahlpflichtfach Zertifikat IT-Sicherheit |
| Literatur |
| <ul style="list-style-type: none"> • Kennedy D., O'Gorman J., Kearns D., Ahoroni M.: Metasploit: The Penetration Tester's Guide, no starch press 2011, ISBN-13: 9781593272883 • Erickson J.: Hacking, Die Kunst des Exploits. 2009, no starch press, 2004, ISBN 10:1593270070 • D.Bernstein, Buchmann, J.: Post-Quantum Cryptography, Springer 2008 |
| Modulverantwortlicher/Modulverantwortliche |
| P. Felke |

| |
|----------------|
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|----------------|--------|
| Digitaltechnik | 2090 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 4 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Hardwaregrundlagen |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Digitaltechnik (3 SWS): D. Rabe • Praktikum Digitaltechnik (1 SWS): D. Rabe |
| Lehrinhalte |
| <p>Stichworte zum Vorlesungsinhalt:</p> <ol style="list-style-type: none"> 1. Digitale Signale: Wert-/Zeit-Diskretisierung, Graycode; 2. Integrierte Schaltungen (CMOS): Analyse von CMOS-Gattern (Transistorschaltung), Modellierung von CMOS-Schaltungen mit Switch-Level Modell; 3. Bussysteme: I2C und V24-Schnittstelle; 4. Schaltnetze (Minimierung: Multi-Output-Minimierung und Quine-McCluskey-Verfahren); Schaltwerke (Hardware-Automaten: Moore- und Mealy-Automaten); 5. Schieberegister: Anwendungen, rückgekoppelte Schieberegister (Fibonacci- und Galois), Cyclic Redundancy Check, mathematische Modellierung als Mod-2 Division; 6. Architekturen Arithmetischer Einheiten am Beispiel von Addierer-Architekturen; 7. Einführung VHDL (Prinzip der Nebenläufigkeit, Aufbau einer VHDL-Beschreibung (Entity, Architecture, strukturelle und Verhaltensbeschreibungen, nenbläufige Signalzuweisungen und Prozesse(sequentiell und kombinatorisch), CAD-Werkzeuge zur Schaltungssynthese, FPGA-Synthese); 8. Testen integrierter Schaltungen: D-Algorithmus; 9. Speicher (SRAM, DRAM, ROM, EEPROM, Flash); <p>Im Praktikum werden die Lehrinhalte durch praktische Aufgaben zu Addiererarchitekturen, Automaten, VHDL, rückgekoppelten Schieberegistern und der Analyse mittels Logic Analyzer vertieft.</p> |

| |
|--|
| angestrebte Lernergebnisse |
| Die Studierenden |
| <ul style="list-style-type: none">• erläutern die Problematiken, die bei der Wertdiskretisierung entstehen können und wie diese Probleme durch die Verwendung des Graycodes vermieden werden können. Sie sind in der Lage zu Dezimal-/Dualzahlen zugehörigen Graycodes und anders herum zu berechnen.• bestimmen die Booleschen Funktion zu CMOS-Transistorschaltungen und modellieren das Verhalten mit dem Switch-Level Modell, das zur Deutung von Verzögerungszeiten und der Verlustleistung in CMOS-Schaltungen verwendet werden kann.• analysieren und synthetisieren serielle I2C- und V24-Übertragungsprotokolle und bestimmen die Signalverläufe zu übertragenen Daten/Adressen. Sie erklären die Konzepte zu Protokollerweiterungen des I2C-Protokolls (10-Bit-Addressierung), die Realierung als wired-and-Topologie sowie des Multi-Master-Betriebs.• sind in der Lage, im Bereich der Schaltungssynthese Boolesche Minimierungen mit Hilfe der Multi-Output-Minimierung (KV-Minimierung) und des Quine-McCluskey-Verfahrens zu berechnen (disjunktive und konjunktive Minimalformen).• entwickeln Hardware-Automaten (Moore-/Mealy) zu einer verbal formulierten Schaltungsssteuerung durch Verwendung der Standard-Entwurfsschritte (Zustandsfolgediagramm, Zustandskodierung, Zustandsfolgetabelle, ggf. Zustandsminimierung, Minimierung der Zustandsfolge- und Ausgabefunktionen, Umsetzung der Hardware-Schaltung) und sind in der Lage, das zeitliche Verhalten der Automaten händisch zu simulieren und die unterschiedlichen Verhaltensmuster von Moore/Mealy-Automaten gegenüberzustellen.• benennen die Anwendungen von Schieberegisterstrukturen und identifizieren die unterschiedlichen rückgekoppelten Schieberegisterstrukturen (Moore/Mealy) anhand von Signalverläufen; sie berechnen die Signalverläufe per Mod-2-Division und bestimmen/berechnen Prüfmuster für Cyclic-Redundancy-Check-Module.• analysieren einfache digitale Schaltungen, die in VHDL beschrieben sind und führen das Prinzip der Nebenläufigkeit als Unterschied zu Software-Programmiersprachen aus.• berechnen Testmustern zum Produktionstest integrierter digitaler Schaltungen (D-Algoritmus, Scan-Ketten).• legen die Prinzipien unterschiedlicher Speicher mit Schwerpunkt auf Halbleiterspeichern dar. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Studiengangschwerpunkte |
| Wahlpflichtmodul Zertifikat Technische Informatik |
| Literatur |
| Woitowitz, R., Urbanski, K.: Digitaltechnik: Ein Lehr- und Übungsbuch, Springer-Verlag; D. Rabe: Digital- und Mikroprozessortechnik (Online-Modul für das entsprechende Online-Modul, das den Studierenden frei zur Verfügung gestellt wird); weitere Folien mit Begleitvideos |
| Modulverantwortlicher/Modulverantwortliche |
| D. Rabe |
| Verwendbarkeit |
| BI, BET, BETPV, BIPV |

| Modulname | Nummer |
|----------------------|--------|
| Eingebettete Systeme | 2100 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Rechnerorganisation, Hardwarenahe Programmierung, Digitaltechnik |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Eingebettete Systeme (2 SWS): G. von Cölln • Praktikum Eingebettete Systeme (2 SWS): G. von Cölln |
| Lehrinhalte |
| Der Aufbau und die Funktionen von aktuellen Mikrocontrollern sowie deren Konzepte zur Programmierung in einer Hochsprache mit modernen Entwicklungsmethoden werden vorgestellt. Die Programmierung peripherer Baugruppen wird exemplarisch eingeführt und an praktischen Aufgabenstellungen verdeutlicht. |
| angestrebte Lernergebnisse |
| Die Studierenden verfügen über ein fundiertes, anwendungsorientiertes Wissen über den Aufbau, die Arbeitsweise und die Programmierung moderner Mikrocontroller. Sie sind in der Lage die Leistungsfähigkeit von Mikrocontrollern zu beurteilen und kennen das Zusammenwirken von Hardware- und Software. Die Studierenden sind mit der Funktion und Programmierung peripherer Baugruppen vertraut. Sie kennen aktuelle Entwicklungswerkzeuge und -methoden und können ihr Wissen zur Lösung von praxisnahen Aufgabenstellung in Gruppenarbeiten anwenden. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Studiengangsschwerpunkte |
| Wahlpflichtfach Zertifikat Technische Informatik |
| Literatur |
| R. Toulson, Fast and Effective Embedded Systems Design: Applying the ARM mbed, Newnes, 2016 E. White, Making Embedded Systems, O'Reilly, 2011 G. Dean, Embedded Systems Fundamentals with Arm Cortex-M bases Microcontrollers, arm Education Media, 2017 |
| Modulverantwortlicher/Modulverantwortliche |
| G. von Cölln |

| |
|----------------------|
| Verwendbarkeit |
| BI, BET, BETPV, BIPV |

| Modulname | Nummer |
|----------------|--------|
| HW/SW Codesign | 2110 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung oder Studienarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| Hardwarenahe Programmierung |
| Empfohlene Voraussetzung |
| Digitaltechnik, Eingebettete Systeme, Hardwareentwurf mit VHDL |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • HW/SW Codesign (2 SWS): C. Koch • Praktikum HW/SW Codesign (2 SWS): C. Koch |
| Lehrinhalte |
| Die Vorlesung HW/SW Codesign behandelt typische Zielarchitekturen und HW/SW-Komponenten von eingebetteten Standard-Systemen und System-on-Programmable-Chips (SoPC) sowie deren Entwurfswerzeuge für ein Hardware/Software Codesign. Hierbei behandelte Zielarchitekturen und Rechenbausteine umfassen Mikrocontroller, DSP (VLIW, MAC), FPGA, ASIC, System-on-Chip als auch hybride Architekturen. Weitere Stichworte sind: Hardware/Software Performanz, Sequentielle oder parallele Verarbeitung, Multiprozessorsysteme (UMA, NUMA, Cache-Kohärenz), Custom Instruction, Custom Peripherals, IP-Core (Soft-IP-Core, Hard-IP-Core) und Bus-Konzepte eingebetteter Systeme (Gateway, Bridge, Marktübersicht). |
| angestrebte Lernergebnisse |
| Ziel der Veranstaltung ist die Zusammenführung der zunächst im Studium getrennten Betrachtung von Hardware- und Software-Systemen zum Aufbau, Entwurf und Analyse moderner eingebetteter Systeme. Die Studierenden haben hierbei weiterführende Kenntnisse bezüglich eingebetteter Systeme als auch deren Partitionierung erworben und beherrschen grundlegende Methoden zum Design und zur Programmierung eines System-on-Programmable-Chips (SoPC). |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Studiengangschwerpunkte |
| Wahlpflichtfach Zertifikat Technische Informatik |
| Literatur |
| Schaumont, P.: A Practical Introduction to Hardware/Software Codesign, Springer, 2013 Mahr, T.: Hardware-Software-Codesign, Vieweg Verlag Wiesbaden, 2007. Patterson, D.A.: Rechnerorganisation und -entwurf, Elsevier München, 2005 |

| |
|--|
| Modulverantwortlicher/Modulverantwortliche |
|--|

| |
|---------|
| C. Koch |
|---------|

| |
|----------------|
| Verwendbarkeit |
|----------------|

| |
|----------------------|
| BI, BET, BETPV, BIPV |
|----------------------|

| Modulname | Nummer |
|--------------------------|--------|
| Hardwareentwurf mit VHDL | 2120 |

| | |
|------------------------------|---|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Test am Rechner oder Klausur oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Digitaltechnik |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Hardwareentwurf mit VHDL (2 SWS): D. Rabe • Praktikum Hardwareentwurf mit VHDL (2 SWS): D. Rabe |
| Lehrinhalte |
| <p>Stichworte zum Vorlesungsinhalt:</p> <ul style="list-style-type: none"> • Hardwarebeschreibungssprache VHDL • synthetisierbarer VHDL-Code; • Schaltungssynthese (Synthese, STA); • Schaltungssimulation (Testbench); Im Praktikum werden diese Lehrinhalte durch entsprechende Aufgaben vertieft. |
| angestrebte Lernergebnisse |
| <p>Die Studierenden kennen und verstehen die Beschreibung sowie Simulation digitaler Schaltungen mit VHDL. Hierbei werden digitale Schaltungen bewusst in kombinatorische (Schaltnetze) und sequentielle Schaltungsteile (Schaltwerke) zergliedert. Die Studierenden verwenden VHDL zur Realisierung von Automaten, rückgekoppelten Schieberegistern, arithmetischen Einheiten sowie der Ansteuerung von SRAM-Speichern. Sie kennen und verstehen außerdem die Umsetzung dieser Beschreibungen in eine FPGA-basierte Hardwareimplementierung mit den entsprechenden CAD-Werkzeugen. Hierzu gehört insbesondere die simulationsbasierte Verifikation der mit VHDL beschriebenen digitalen Schaltungen und die Durchführung der timing-driven Synthese sowie der statischen Timinganalyse.</p> |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Studiengangschwerpunkte |
| Wahlpflichtfach Zertifikat Technische Informatik |
| Literatur |
| Ashenden, P.: The Designer's Guide to VHDL, Morgan Kaufmann Publishers, 2008 |

| |
|--|
| Modulverantwortlicher/Modulverantwortliche |
|--|

| |
|---------|
| D. Rabe |
|---------|

| |
|----------------|
| Verwendbarkeit |
|----------------|

| |
|----------------------|
| BI, BET, BETPV, BIPV |
|----------------------|

| Modulname | Nummer |
|----------------|--------|
| Computergrafik | 2130 |

| | |
|------------------------------|--|
| ECTS | 7,5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 3 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 90 h Kontaktzeit + 135 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur (1,5 h) oder mündliche Prüfung (0,5 h) oder Kursarbeit (Erstellung digitaler Medien) |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• Computergrafik (4 SWS): I. Schebesta• Praktikum Computergrafik: I. Schebesta |
| Lehrinhalte |
| Rastergrafik, Vektorgrafik, Bézier-Kurven, 3D-Grafik, Farbtheorie, Wahrnehmungstheorie, Grafikformate, Kompression, Fraktale, iterative Systeme, Visualisierung, Transformationen, Projektion, Betrachtungspyramide, Farbtemperatur, HDRI, plenoptische Funktion, Koordinatensysteme, Augmented Reality, künstliche Intelligenz, ethische Relevanz von Bildmanipulationen. |
| angestrebte Lernergebnisse |
| Die Studentinnen und Studenten kennen die wesentlichen Grundlagen Computergrafik. Sie können diese Kenntnisse bei entsprechenden Problemstellungen in den Ingenieurwissenschaften praxis- bzw. anwendungsbezogen einsetzen. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| Computergrafik, 3. Auflage, Wiesbaden, Vieweg+Teubner, 2011. Nischwitz, Alfred et |
| Modulverantwortlicher/Modulverantwortliche |
| I. Schebesta |
| Verwendbarkeit |
| Bl, BIPV, BMT |

| Modulname | Nummer |
|-------------------------|--------|
| Drahtlose Sensortechnik | 2140 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5h oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Rechnerorganisation |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Drahtlose Sensortechnik (2 SWS): G. von Cölln • Praktikum Drahtlose Sensortechnik (2 SWS): G. von Cölln |
| Lehrinhalte |
| Grundlegender Aufbau von IoT-Devices und Sensoren, Energiemessung, Mikrocontroller und Sensoren, Energieaufnahme und -optimierung, Kommunikation, Energy-Harvester und Energieversorgung |
| angestrebte Lernergebnisse |
| Die Studierenden verstehen grundlegende Konzepte aus dem Bereich der drahtlosen Sensorsysteme. Auf der Grundlage dieses Wissens ordnen Sie Anforderungen verschiedener Nutzergruppen fachgerecht den vermittelten Konzepten zu. Die Studierenden können selbstständig Systemarchitekturen für drahtlose Sensoren erstellen, optimieren und evaluieren. Insbesondere werden Verfahren zur Analyse und Optimierung der Verlustleistung behandelt, die die Verwendung von Energy-Harvestern ermöglichen. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Literatur |
| Klaus Dembowski, Energy Harvesting für die Mikroelektronik, VDE Verlag Mauri Kuorilehto, Ultra-Low Energy Wireless Sensor Networks in Practice, Wiley, 2007 |
| Modulverantwortlicher/Modulverantwortliche |
| G. von Cölln |
| Verwendbarkeit |
| Bl, BET, BETPV, BIPV |

| Modulname | Nummer |
|-----------|--------|
| Englisch | 2150 |

| | |
|------------------------------|--|
| ECTS | 2,5 |
| Semesterwochenstunden | 2 |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 30 h Kontaktzeit + 120 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1h |

| |
|---|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Einstiegsniveau entsprechend dem gewünschten Qualifikationsziel, z.B. CEF A2 erforderlich für CEF B1 nach 2 Semestern |

| |
|--|
| Lehrveranstaltungen |
| Englisch (2 SWS): M. Parks |
| Lehrinhalte |
| Grammatik Wiederholung und praktische Aufgaben. Einführung und Nutzung von Vokabular, Ausdrücken und grammatischen Ausdrucksweisen. Gezielte Ausbildung von Fähigkeiten: Beschreibung, Erklärung, Analyse und Vergleiche von Komponenten, Systemen und Prozessen. Spezifizieren von Anforderungen; Formulierung von Fragen. Ausdrücken von Meinungen, Zustimmungen und Ablehnungen. Ausdrücken von Absichten; Festlegen von Planungen; Anbieten von Empfehlungen. Erteilen, Interprätieren und Ausführen von Instruktionen. Verstehen und beschreiben von Ursache und Wirkung. |
| angestrebte Lernergebnisse |
| CEF Levels (sprachlich und schriftlich): A2 -- CEF-B1 B1 -- CEF-B2 B2 -- CEF-C1 |
| Lehr- und Lernmethoden |
| Auf der Basis von CEF-Levels (Common European Framework): 1. Lektionen/Veranstaltungen zu speziellen Themen für Arbeiten im Technischen Umfeld 2. Intensives Sprechen, Zuhören und Schreiben mit laufenden Feedback 3. Diskussionen und Rollenspiele 4. Regelmäßige kurze Fortschrittsteste mit Feedback 5. Schriftliche Abschlußprüfung |
| Literatur |
| Technical English (Pearson); ausgewählte Texte aus Fachschriften und websites. |
| Modulverantwortlicher/Modulverantwortliche |
| M. Parks |
| Verwendbarkeit |
| BI, BET, BETPV, BIPV, BMD, BMDPV, BMT, BWEM |

| Modulname | Nummer |
|----------------------------|--------|
| Kalkulation und Teamarbeit | 2170 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung 0,5 h oder Kursarbeit ca. 20 Seiten |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Kalkulation und Angebotserstellung (2 SWS): L. Jäncchen • Teamarbeit und angewandtes Projektmanagement (2 SWS): L. Jäncchen |
| Lehrinhalte |
| <p>Drei Ansätze zur Preisfindung: Kundenorientiert Kosteorientiert Wettbewerbsorientiert Aufbau von Angeboten im B2B Umfeld Ausrichtung von Angeboten auf individuelle kunden-spezifische Bedürfnisse Ausbau und Organisation von Teamarbeit Kritische Erfolgsfaktoren Ursachen für Probleme angestrebte Lernergebnisse</p> <p>Studierende können für technische Anlagen oder für technische Produkte Preise vorschlagen und branchenübliche Angebote verfassen. Weiter begreifen Sie Arbeit im Marketing und Vertrieb als Teamarbeit und können diese strukturieren und organisieren. Dafür wenden Sie verschiedene Ansätze zur Preiskalkulation an und setzen in der Analyse der Ergebnisse Preise fest. Die Studierenden kennen den prinzipiellen Aufbau von Angeboten im B2B Bereich und formulieren kundenspezifische Angebote, indem Sie die jeweils spezifischen Bedürfnisse des Kunden individuell adressieren. Weiter kennen die Studierenden wesentliche Erfolgsfaktoren für ein Gelingen sowie typische Gründe für ein Scheitern von Teamarbeit und können in der Berücksichtigung dessen Team organisieren, strukturieren und Projekte managen. Studierende bringen sich bewusst in Teams ein und leisten einen signifikanten Beitrag zum Teamerfolg. Dies ermöglicht Studierenden insbesondere im B2B Bereich Preise zu bestimmen, Angebote zu verfassen und effizient in Team zu arbeiten.</p> |
| Lehr- und Lernmethoden |
| Vorlesung, Übungen |
| Studiengangschwerpunkte |
| Wahlpflichtfach Zertifikat Marketing und Vertrieb |

| |
|---|
| Literatur |
| Schmidt, A.: Kostenrechnung; 5. Aufl., Stuttgart 2009 |
| Meier, Rolf.: Erfolgreiche Teamarbeit. In: Gabal Verlag GmbH, Offenbach (2006) ISBN 3-89749-585-6 |
| Modulverantwortlicher/Modulverantwortliche |
| L. Jäncchen |
| Verwendbarkeit |
| BET, BETPV, BI, BIPV, BMT |

| Modulname | Nummer |
|---|--------|
| Kommunikation in Marketing und Vertrieb | 2180 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung 0,5 h oder Kursarbeit ca. 20 Seiten |

| Voraussetzung laut Prüfungsordnung für die Teilnahme |
|--|
| keine |
| Empfohlene Voraussetzung |
| keine |

| Lehrveranstaltungen |
|--|
| Kommunikation in Marketing und Vertrieb (4 SWS): L. Jänchen |
| Lehrinhalte |
| Studierende wenden Sie die Grundregeln des klassischen Verhandelns nach dem Harvard-Konzept an und können rhetorische Methoden gezielt einsetzen. |
| angestrebte Lernergebnisse |
| Die Studierenden lernen verschiedene typische Kommunikationssituationen in Marketing und Vertrieb kennen. Sie entwickeln ein klares Verständnis für die Spezifika der jeweiligen Kommunikation. Sie sind in der Lage sich entsprechend vorzubereiten und in der Kommunikation ihr Verhalten auf die jeweilige Situation abzustimmen. So können sich Studierende systematisch auf Verhandlungen vorbereiten, diese planen und durchführen. Weiter können sie rhetorische Instrumente anwenden, um verschiedene Gesprächs- und Verhandlungssituationen zu steuern, insbesondere in Verhandlungen, in der Präsentation eigener Ideen und in Vertriebsgesprächen. Dazu wenden Studierende die Grundregeln des klassischen Verhandelns nach dem Harvard-Konzept an und können rhetorische Methoden gezielt einsetzen. Dies ermöglicht ihnen Win-Win Verhandlungsergebnisse zu erzielen sowie in Verhandlungen, in Vertriebsgesprächen und allgemein Situation effektiv zu kommunizieren. |
| Lehr- und Lernmethoden |
| Vorlesung (mit Übungen) |
| Studiengangschwerpunkte |
| Wahlpflichtfach Zertifikat Marketing und Vertrieb |

Literatur

Fischer, Roger; Ury, William; Patton, Bruce: Das Harvard-Konzept, In: Campus Verlag, Frankfurt/New York (2006), ISBN 978-3-593-38135-0

Heinz M. Goldmann: Wie man Kunden gewinnt: Cornelsen Verlag, Berlin (2002), ISBN 3-464-49204-4

Kohlert, H.: Internationales Marketing für Ingenieure

Modulverantwortlicher/Modulverantwortliche

L. Jänchen

Verwendbarkeit

BET, BETPV, BI, BIPV, BMT

| Modulname | Nummer |
|--------------------------|--------|
| Marketing für Ingenieure | 2190 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung 0,5 h oder Kursarbeit ca. 20 Seiten |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Marketing für Ingenieure (2 SWS): L. Jänen • Praktikum Marketing für Ingenieure (2 SWS): L. Jänen |
| Lehrinhalte |
| Einordnung des Marketing in das Unternehmen, Einführung in den B2B Kaufprozess, eine Einführung in ausgewählte, häufig angewandte Methoden des Marketing und Produktmanagements, Definition von Zielkunden und Erhebung derer Probleme und Bedürfnisse, Definition von Produkten als Problemlösungen, Grundlagen von Marketingstrategien und der Elemente des Marketingmix sowie ein Überblick über Marketingorganisation und -kontrolle. |
| angestrebte Lernergebnisse |
| Die Studierenden können einfache Marketingkonzepte für technische Produkte entwickeln und überzeugend darstellen. Dafür analysieren Sie Anwender-/Kundenprobleme, die Markt- und die Wettbewerbssituation sowie Aspekte der Wirtschaftlichkeit und Nachhaltigkeit und definieren darauf aufbauend Produkte als Problemlösungen. Sie entwickeln Marketingstrategien und entwerfen Maßnahmen im Marketing-Mix zur deren Umsetzung und präsentieren Ihre Konzepte. Dies ermöglicht den Studierenden mit Ihrem Denken auf der Schnittstelle von Technik und Marketing nicht nur technisch machbare sondern auch relevante, nachhaltige und kommerziell erfolgreichere Produkte als Problemlösung zu entwerfen zu entwickeln und zu vermarkten. |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum, Studentische Arbeit |
| Studiengangschwerpunkte |
| Wahlpflichtfach Zertifikat Marketing und Vertrieb |
| Literatur |
| Kohlert, H.: Marketing für Ingenieure mit vielen spannenden Beispielen aus der Unternehmensspraxis, Oldenbourg Verlag, 3. Auflage 2013 Bruhn, M.: Marketing -- Grundlagen für Studium und Praxis. Gabler, 9. Auflage, 2008 |

Modulverantwortlicher/Modulverantwortliche

L. Jänchen

Verwendbarkeit

BET, BETPV, BI, BIPV, BMT

| Modulname | Nummer |
|-------------------|--------|
| Mediendramaturgie | 2200 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 5 (jedes Wintersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Kursarbeit (ca. 20 Seiten) und/oder Referat (15 Min) |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| Mediendramaturgie (4 SWS): I. Schebesta |
| Lehrinhalte |
| Dramaturgie, Komödie, Drama, Aufbau von Geschichten, Konflikte, Handlungskonstruktion, Exposition, Spannungsbögen, Katharsis, Protagonisten, Antagonisten, Figurenentwicklung, Wendepunkte, Nebenhandlung, Drei-Akt-Schema, Fünf-Teile-Schema, Heldenreise, Dialoge, Drehbuchformen, etc. |
| angestrebte Lernergebnisse |
| Erkennen, aus welchen Elementen eine Geschichte besteht. Lernen, wie man Spannung aufbaut. Wissen über das technische Handwerkzeug eines Drehbuchautors und seiner Arbeitsweisen. |
| Lehr- und Lernmethoden |
| Seminar, studentische Arbeit, Vortrag |
| Literatur |
| Aristoteles: Poetik, Independently published, 2021. Kerstin Stutterheim: Handbuch angewandter Dramaturgie, Peter Lang Verlag, 2015. Gustav Freytag: Die Technik des Dramas, Forgotten Books, Berlin 2018. Christopher Vogler: Die Odyssee der Drehbuchschreiber, Romanautoren und Dramatiker: Mythologische Grundmuster für Schriftsteller, Autorenhaus-Verlag, Berlin 2018. Syd Field: Das Drehbuch, Autorenhaus Verlag GmbH, 2007. Linda Seger: Von der Figur zum Charakter, Alexander Verlag, Berlin 2012. |
| Modulverantwortlicher/Modulverantwortliche |
| I. Schebesta |
| Verwendbarkeit |
| BMT, BET, BETPV, BI, BIPV |

| Modulname | Nummer |
|---------------|--------|
| Mixed Reality | 2210 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Studienarbeit |

| Voraussetzung laut Prüfungsordnung für die Teilnahme |
|---|
| Computeranimation, Computergrafik, Interaktive Systeme 2 BaMT: Computeranimation, Computergrafik, Interaktive Medien 2 Bal: Computergrafik, Programmieren 2, Modellierung BalP: Computergrafik, Grundlagen der Programmierung 2, Softwaretechnik |
| Empfohlene Voraussetzung |
| keine |

| Lehrveranstaltungen |
|---|
| <ul style="list-style-type: none">• Mixed Reality (2 SWS); T. Pfeiffer• Praktikum Mixed Reality (2 SWS); T. Pfeiffer |
| Lehrinhalte |
| <p>Theorie</p> <ul style="list-style-type: none">• Grundlagen zu Augmented Reality und Virtual Reality• Methoden zur Entwicklung echtzeitfähiger Mixed-Reality-Anwendungen• Design-Prinzipien für immersive Medien, am Beispiel von Storytelling und Instructional Design• Aufbau von VR/AR Anwendungen (Szenengraph, Datengraph, Renderloop) <p>Praktischer Teil</p> <ul style="list-style-type: none">• Konzeption einer immersiven Erfahrung für Augmented Reality oder Virtual Reality• Produktion der echtzeitfähigen Medien (360°-Film, 3D Modelle, Animationen, etc.)• Entwicklung eines Prototyps einer Mixed-Reality-Anwendung <p>Die grundlegenden Lehrinhalte werden in Vorlesungsform vermittelt und im Rahmen des Praktikums umgesetzt.</p> |
| |

angestrebte Lernergebnisse

Die Studierenden können Mixed-Reality-Technologien (Virtual Reality, Augmented Reality) einsetzen, um interaktive Erfahrungen zu entwickeln. Dazu berücksichtigen sie die Grundlagen der unterschiedlichen Mixed-Reality-Technologien und gestalten Inhalte entsprechend der technischen Herausforderungen. In der Umsetzung orientieren sie sich entsprechend der Ausrichtung der interaktiven Erfahrungen an den jeweiligen Grundprinzipien der Gattung (z.B. Instructional Design und Immersive Storytelling für digitale Trainings).

Konkret können die Studierenden

- Grundlagen zu Augmented Reality und Virtual Reality verstehen,
- Inhalte für Mixed-Reality-Technologien konzipieren und umsetzen,
- Interaktive Erfahrungen prototypisch entwickeln.

Lehr- und Lernmethoden

Vorlesung und Seminar

Studiengangschwerpunkte

Wahlpflichtfach Zertifikat Computer-Aided Media Production und Zertifikat Virtuelle Welten

Literatur

Dörner, R.; Broll, W.; Grimm, P.; Jung, B.: Virtual und Augmented Reality (VR/AR): Grundlagen und Methoden der Virtuellen und Augmentierten Realität. Springer Verlag, 2. Auflage, 11. Oktober 2019.

Modulverantwortlicher/Modulverantwortliche

T. Pfeiffer

Verwendbarkeit

BMT, BI, BIPV

| Modulname | Nummer |
|-----------------------------|--------|
| Produktion Digitaler Medien | 2220 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Kursarbeit (Erstellung digitaler Medien) |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| Produktion digitaler Medien (4 SWS): I. Schebesta |
| Lehrinhalte |
| Mögliche Digitale Medien wären z.B. die folgenden: Animation(2D,3D), Interaktive Medien (Unity 3D), Visuelle Effekte/Compositing, Technik des Drehbuchschreibens, Möglichkeiten des eBooks, Bewegtbild/Film, Filmbeitrag (1:30), Erklär-Film, Kurz-Portrait (einer Person), Fake-Documentary, Internet-Video-Serie, alte und neue Sendeformate, Experimentelles, Unterhaltung/Komik, Zeitraffer-Aufnahmen, Stereofilm, Virtuelle Realität, Videospiel, Motion Capturing, fiktive Person in sozialen Medien einschleusen (wie bei LonelyGirl), HOAX generieren, Hörspiel, digitale Kunst, interaktive Exponate, Projection-Mapping |
| angestrebte Lernergebnisse |
| Die Teilnehmer kennen neue Möglichkeiten der Produktion von digitalen Medien. Sie können im Team selbstständig ein digitales Medium konzeptionieren und produzieren. |
| Lehr- und Lernmethoden |
| Vorlesung, Studentische Arbeit |
| Studiengangsschwerpunkte |
| Wahlpflichtfach Zertifikat Computer-Aided Media Production |
| Literatur |
| Dinur, Eran: "The Filmmaker's Guide to Visual Effects: The Art and Techniques of VFX for Directors, Producers, Editors and Cinematographers", Routledge, 2017. Borromeo, Nicolas Alejandro: Hands-On Unity 2021 Game Development: Create, customize, and optimize your own professional games from scratch with Unity 2021, 2nd Edition, Packt Publishing, 2021. |
| Modulverantwortlicher/Modulverantwortliche |
| I. Schebesta |

| |
|----------------|
| Verwendbarkeit |
| BMT, BI, BIPV |

| Modulname | Nummer |
|---------------------------------|--------|
| Spezielle Themen der Informatik | 2230 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 4 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung oder Kursarbeit |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| Spezielle Themen der Informatik (4 SWS): Lehrende der Abteilung E+I |
| Lehrinhalte |
| Werden den Studierenden vor Beginn der Veranstaltung bekanntgegeben. |
| angestrebte Lernergebnisse |
| Werden den Studierenden vor Beginn der Veranstaltung bekanntgegeben. |
| Lehr- und Lernmethoden |
| Vorlesung oder Praktikum oder Seminar |
| Literatur |
| Werden den Studierenden vor Beginn der Veranstaltung bekanntgegeben. |
| Modulverantwortlicher/Modulverantwortliche |
| Studiendekan |
| Verwendbarkeit |
| BI, BIPV |

| Modulname | Nummer |
|----------------------|--------|
| Systemprogrammierung | 2240 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Studienarbeit oder mündliche Prüfung |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Betriebssysteme, C/C++ |

| |
|--|
| Lehrveranstaltungen |
| Systemprogrammierung (4 SWS): C. Link |
| Lehrinhalte |
| Folgende Themen werden behandelt: Am Beispiel von Linux/Unix werden die Basisideen und Konzepte der gängigen Dateisysteme, der TCP/IP-basierten Netzwerkdienste sowie der Verwaltung von Geräten und Prozessen dargestellt. Moderne APIs zur effizienten Abarbeitung von Hochleistungs-I/O und zur Kernel-Anbindung bzw. Überwachung werden behandelt und in Prototypen verwendet. |
| angestrebte Lernergebnisse |
| Die Studierenden sind in der Lage Rechnersysteme mit Hilfe von Skripten zu installieren, zu konfigurieren, zu verwalten und Leistungsmessungen durchzuführen, so dass die zu verwaltenden Rechner den jeweiligen Anforderungen optimal entsprechen. Die Studierenden können System- und Kernel-nahe APIs einsetzen, um Lösungen für besondere Anwendungsbereiche zu entwickeln. |
| Lehr- und Lernmethoden |
| Seminar |
| Literatur |
| Kerrisk, M.: The Linux Programming Interface: A Linux and UNIX System Programming Handbook, No Starch Press 2010 Rago, S. A., Stevens, W. R.: Advanced Programming in the UNIX Environment, Addison Wesley 2013 |
| Modulverantwortlicher/Modulverantwortliche |
| C. Link |
| Verwendbarkeit |
| BI, BET, BETPV, BIPV |

| Modulname | Nummer |
|-------------------|--------|
| Vertriebsprozesse | 2250 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Klausur 1,5 h oder mündliche Prüfung 0,5 h oder Kursarbeit ca. 20 Seiten |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|--|
| Lehrveranstaltungen |
| <ul style="list-style-type: none"> • Vertriebsprozesse (2 SWS): L. Jäncchen • Praktikum Vertriebsprozesse (2 SWS): L. Jäncchen |
| Lehrinhalte |
| <p>Analyse der Zielkunden Definition einer Persona Beschreibung des „Customer Journey“ auf dem Weg von der ersten Kontaktaufnahme bis zum Kauf und darüber hinaus Identifikation der Kundenwünsche, -bedürfnisse und -fragen auf dem Customer Journey Entwurf von Prozessschritten zur Unterstützung des Customer Journey Funktionalität von CRM-Systemen</p> |
| angestrebte Lernergebnisse |
| <p>Studierende verstehen den Vertrieb als Abfolge systematischer, integrierter und strukturierter Prozesse. Sie können derartige Prozesse unter Berücksichtigung der jeweiligen Wünsche und Bedürfnisse der Zielkunden definieren, aktiv ausgestalten und durchlaufen. Dazu analysieren Sie die jeweiligen Wünsche, Bedürfnisse und Fragen der Zielkunden auf deren Weg von der ersten Kontaktaufnahme über den Kauf und darüber hinaus und entwerfen Prozesse zur Befriedigung und Beantwortung. Sie gliedern dabei die Prozesse in die Phasen „Find“, „Win“ und „Keep“. Studierende erkennen die Bedeutung und Möglichkeiten von modernen CRM-Systemen zur Unterstützung und partiellen Automatisierung dieser Prozesse. Dies ermöglicht den Studierenden einen effektiven zielkundenspezifischen Vertrieb in Grundelementen zu planen und zielgerichtet auch durch die Verwendung moderner CRM-Systeme vertrieblich zu arbeiten.</p> |
| Lehr- und Lernmethoden |
| Vorlesung, Praktikum |
| Studiengangsschwerpunkte |
| Wahlpflichtfach Zertifikat Marketing und Vertrieb |

Literatur

DWECK, Carol S., PH.D.: Mindset, In: Random House, Inc., New York (2006)
Peoples, David: Selling to The Top, In: Wiley&Sons, Canada (1993), ISBN 0-471-58104-6
Homburg, Schäfer, Schneider: Sales Excellence, 6. Auflage, Gabler Verlag, 2011, ISBN 978-3-8349-2279-3

Modulverantwortlicher/Modulverantwortliche

L. Jänchen

Verwendbarkeit

BET, BETPV, BI, BIPV, BMT

| Modulname | Nummer |
|------------------|--------|
| Visuelle Effekte | 2260 |

| | |
|------------------------------|--|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Dauer | 1 Semester |
| Semester (Häufigkeit) | 6 (jedes Sommersemester) |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Kursarbeit (Erstellung digitaler Medien) |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| Visuelle Effekte (4 SWS): B. Arp (LB) |
| Lehrinhalte |
| 2D- und 3D-Compositing, 2D- und 3D-Tracking, Match Moving, Greenscreen-Verfahren, In-Camera-Effekte, Matte-Effekte, Postprocessing-Effekte, modellbasierte Effekte, Überblend-Effekte, HDR-Fotografie zum Einsatz für global Illumination. Motion-Capturing, virtual production with LED video walls. |
| angestrebte Lernergebnisse |
| Die Teilnehmer können mit einer Compositingsoftware sowie einer 3D-Animationssoftware umgehen. Sie können einen Special-Effekt analysieren, planen und durchführen. Die Teilnehmer durchschauen, wie moderne, mit dem Computer erzeugte Effekte auf historisch gewachsener Tricktechnik der Filmindustrie fußen. |
| Lehr- und Lernmethoden |
| Seminar |
| Literatur |
| Mulack, Thomas; Giesen, Rolf: "Special Visual Effects - Planung und Produktion", Bleicher Verlag, 2002 Dodds, David: "Motion Graphic Design with Adobe After Effects 2022 - Second Edition: Develop your skills as a visual effects and motion graphics artist", Packt Publishing, 2022. Brinkmann, Ron: "The Art and Science of Digital Compositing: Techniques for Visual Effects, Animation and Motion Graphics (The Morgan Kaufmann Series in Computer Graphics) 2nd Edition", Morgan Kaufmann, 2008. Dinur, Eran: "The Filmmaker's Guide to Visual Effects: The Art and Techniques of VFX for Directors, Producers, Editors and Cinematographers", Routledge, 2017. |
| Modulverantwortlicher/Modulverantwortliche |
| I. Schebesta |

| |
|----------------|
| Verwendbarkeit |
| BMT, BI, BIPV |

| Modulname | Nummer |
|--------------------|--------|
| iOS-Programmierung | 2270 |

| | |
|------------------------------|---|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Erstellung und Dokumentation von Rechnerprogrammen (20-30 Seiten pro Person) und/oder mündliche Prüfung 0,5 h |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| Programmieren 2 |

| |
|---|
| Lehrveranstaltungen |
| <ul style="list-style-type: none">• iOS-Programmierung (2 SWS): G. J. Veltink• Praktikum iOS-Programmierung (2 SWS): G. J. Veltink |
| Lehrinhalte |
| Swift, das iOS-SDK, die iOS-Entwicklungswerkzeuge, Mobile Design and Architecture Patterns, Application Frameworks, User Interface Design für iOS-Anwendungen, Benutzung der speziellen Features des iPhones/iPads. Als Leitfaden werden die (englischen!) Materialien des Stanford-Kurses von Prof. Paul Hegarty eingesetzt (Spring 2023): https://cs193p.sites.stanford.edu Hinweis: Zur Teilnahme benötigen Sie einen Zugang zu einem aktuellen Mac-Rechner, z.B. ein persönliches MacBook. (Stand 01.01.2024) |
| angestrebte Lernergebnisse |
| Die Studierenden sollen die "iOS"-Plattform und die zugehörigen Werkzeuge kennenlernen und anschließend selbstständig iOS-Programme (Apps) für das iPhone und iPad entwickeln können. Die Ergebnisse sollen im Team erstellt werden und die wissenschaftlichen Ergebnissen sollen präsentiert werden. |
| Lehr- und Lernmethoden |
| Seminar, Praktikum |
| Literatur |
| Apple: The Swift Programming Language (Swift 5.10). [https://docs.swift.org/swift-book/index.html] Apple: Configuring a multiplatform app. [https://developer.apple.com/documentation/Xcode/configuring-a-multiplatform-app-target]. Alle Dokumente befinden sich in der "iOS Developer Library" unter https://developer.apple.com/documentation (Stand 01.01.2024) |
| Modulverantwortlicher/Modulverantwortliche |
| G. J. Veltink |
| Verwendbarkeit |
| BMT, BET, BETPV, BI, BIPV |

| Modulname | Nummer |
|--------------|--------|
| Gamification | 2300 |

| | |
|------------------------------|---------------------------------------|
| ECTS | 5 |
| Semesterwochenstunden | 4 |
| Pflicht/Wahlpflicht | WP |
| Arbeitsaufwand | 60 h Kontaktzeit + 90 h Selbststudium |
| Prüfungsart, -umfang, -dauer | Kursarbeit und Referat |

| |
|--|
| Voraussetzung laut Prüfungsordnung für die Teilnahme |
| keine |
| Empfohlene Voraussetzung |
| keine |

| |
|---|
| Lehrveranstaltungen |
| Gamification (4 SWS): M. Rauschenberger, K. Nagel, B. Arp |
| Lehrinhalte |
| Die Studierenden erarbeiten in Teams einen Anwendungsfall für Gamification (z.B. ein analoger oder digitaler Prototyp oder eine Integration von Gamification in Prozesse). Dabei ist darauf zu achten, dass der Einsatz von Gamification Zielführend umgesetzt wird und gängige Fallstricke und Risiken vermieden werden. In den Lehreinheiten werden neben den Prinzipien der Gamification auch Grundlagen in Human-Centered Design und agiler Projektorganisation vermittelt. |
| angestrebte Lernergebnisse |
| Die Studierenden erwerben ein umfassendes Verständnis des Konzepts der Gamification sowie zentraler Prozesse (z.B. Human-Centered Gamification Process oder der Octalysis-Methode). Sie lernen einzuschätzen, in welchen Fällen der Einsatz von Gamification sinnvoll ist und worauf bei der praktischen Umsetzung geachtet werden sollte. Dieses Verständnis soll anhand eines praktischen interdisziplinären Projektes erarbeitet und vertieft werden. |
| Lehr- und Lernmethoden |
| Seminar |
| Literatur |
| Deterding, S., Sicart, M., Nacke, L., O'Hara, K., & Dixon, D. (2011). Gamification. using game-design elements in non-gaming contexts. Proceedings of the 2011 Annual Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '11, 2425. https://doi.org/10.1145/1979742.1979575 ISO 9241-210:2019, Ergonomics of human-system interaction, Part 210: Human-centred design for interactive systems Schön, E.-M., Buchem, I., Sostak, S., & Rauschenberger, M. (2023). Shift Toward Value-Based Learning: Applying Agile Approaches in Higher Education. In Lecture Notes in Business Information Processing: Vol. 494 LNBIP (pp. 24–41). https://doi.org/10.1007/978-3-031-43088-6_2 Agile Manifesto, https://agilemanifesto.org/ |
| Modulverantwortlicher/Modulverantwortliche |
| M. Rauschenberger |

| |
|---------------------------|
| Verwendbarkeit |
| BI, BET, BETPV, BIPV, BMT |