

Modulhandbuch

für den Studiengang

Bachelor of Science Computational
Engineering (Rechnergestütztes
Ingenieurwesen)

(Prüfungsordnungsversion: 2010)

für das Sommersemester 2025

Inhaltsverzeichnis

Bachelorarbeit (B.Sc. Computational Engineering 2010) (1999).....	5
Computer Science	
Simulation und Wissenschaftliches Rechnen 1 (43370).....	7
Computational Engineering 1 (43610).....	9
Simulation und Wissenschaftliches Rechnen 2 (43871).....	11
Algorithmen und Datenstrukturen (93050).....	13
Systemprogrammierung (93180).....	17
Simulation und Modellierung I (97090).....	21
Mathematics	
Numerik I für Ingenieure (64620).....	25
Numerik II für Ingenieure (64631).....	27
Mathematik für CE 1 (67520).....	28
Mathematik für CE 2 (67530).....	30
Mathematik für CE 3 (67540).....	32
Mathematik für CE 4 (67550).....	34
NF Automatic Control	
Modeling of Control Systems (92241).....	37
Regelungstechnisches Praktikum (92440).....	38
Einführung in die Regelungstechnik (97040).....	39
Regelungstechnik B (Zustandsraummethoden) (97060).....	41
NF Thermo- and Fluidynamics	
Technische Thermodynamik I (92476).....	45
Technische Thermodynamik II (94304).....	47
Wärme- und Stoffübertragung (97030).....	49
NF Information Technology	
Einführung in die Grundlagen der Elektrotechnik für CE-Studierende (43830).....	52
Digitale Signalverarbeitung (93500).....	54
Information Theory and Coding / Informationstheorie und Codierung (93601).....	56
NF Mechatronics	
Elektromagnetische Felder I (92520).....	60
Grundlagen der Elektrotechnik I (92560).....	63
Grundlagen der Elektrotechnik III (92580).....	65
NF Computational Optics	
Modern Optics 1: Advanced Optics (48311).....	68
Experimentalphysik 3: Optik und Quanteneffekte (66120).....	70
Photonik 1 (92390).....	72
Photonik 2 (96350).....	74
NF Solid Mechanics and Dynamics	
Computational Engineering 2 (43840).....	77
Experimentalphysik für Naturwissenschaftler I (66621).....	79
Experimentalphysik für Naturwissenschaftler II (66631).....	81
Dynamik starrer Körper (94500).....	83
Statik, Elastostatik und Festigkeitslehre (94580).....	85
Technische Wahlmodule	
Experimental fluid mechanics (42933).....	88
Physically-based Simulation in Computer Graphics (43385).....	90
Hardware-Software-Co-Design (43490).....	92
Parallele Systeme (43510).....	94
Transportprozesse (43700).....	97
Scientific Visualization (43722).....	99

Kommunikationssysteme (43950).....	101
Introduction to the Finite Element Method (44100).....	103
Pattern Recognition (44130).....	105
Architekturen von Superrechnern (44460).....	108
Cyber-Physical Systems (44470).....	110
Praktische Softwaretechnik (57025).....	112
Introduction to Machine Learning (65718).....	114
Kommunikationsnetze (92290).....	117
Optische Übertragungstechnik (92400).....	119
Komponenten optischer Kommunikationssysteme (92410).....	121
Ereignisdiskrete Systeme (92430).....	123
Nachrichtentechnische Systeme (92601).....	125
Einführung in die moderne Kryptographie (93015).....	128
Parallele und Funktionale Programmierung (93040).....	130
Rechnerkommunikation (93150).....	131
Visualization (93175).....	133
Deep Learning for Beginners (93330).....	135
Digitale Übertragung (93510).....	137
Stochastische Prozesse (93580).....	139
Methode der Finiten Elemente (94550).....	141
Produktionstechnik I und II (94570).....	144
Statik und Festigkeitslehre (94660).....	147
Grundlagen der Elektrischen Antriebstechnik (95010).....	150
Sensorik (92670).....	153
Technische Schwingungslehre (97190).....	155
Mehrkörperdynamik (97270).....	158
Strömungsmechanik II (97330).....	162
Digitale Regelung (97360).....	164
Heterogene Rechnerarchitekturen Online (275245).....	166
Informationsvisualisierung (299892).....	169
Rechnerarchitektur (Vorlesung mit Übung und Rechnerübung) (333815).....	172
High End Simulation in Practice (399607).....	174
Advanced Programming Techniques (465562).....	175
Praktikum Photonik/Lasertechnik 2 (508483).....	177
Künstliche Intelligenz II (532733).....	179
Parallele Systeme (Vorlesung mit erweiterten Übungen) (740665).....	181
Computational Optics CE and MAOT (768903).....	184
Eingebettete Systeme (Vorlesung mit erweiterten Übungen) (773774).....	185
Geometric Modeling (796399).....	188
Rechnerarchitektur (798810).....	191
Mikromechanik (837601).....	193
Künstliche Intelligenz I (894856).....	195
Interactive Computer Graphics (43371).....	197
Computer Graphics Deluxe (43374).....	199
Computer Graphics (43822).....	202
Strömungsmechanik I (97010).....	205
Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung) (292952).....	207
Theoretische Informatik für Wirtschaftsinformatik und Lehramt (93201).....	210
Praktikum CE	
AI-1 Systems Project (93116).....	213
Hackerpraktikum Bachelor (93192).....	215
Supercomputing Praktikum (182798).....	217
Grafik-Praktikum Game Programming (240715).....	219

Praktikum: Entwicklung interaktiver eingebetteter Systeme (unbenotet) (547324).....	220
Industriepraktikum (B.Sc. Computational Engineering 2010) (1995).....	221
Seminar Informatik für CE	
Geschichte der Rechentechnik (47637).....	223
Maschinelles Lernen: Einführung (93127).....	225
Seminar Energieinformatik (93656).....	228
Iterative Lösungsverfahren für lineare und nichtlineare Gleichungssysteme (93096).....	230
Advanced Design and Programming (5-ECTS) (97008).....	232
Seminar Sprachtechnologie für Sprachpathologien (349413).....	234
Seminar Machine Learning Algorithms (628256).....	236
Blender Seminar (921878).....	237
Neuartige Rechnerarchitekturen (941318).....	239
Algorithmen der Simulationstechnik (949119).....	241

1	Modulbezeichnung 1999	Bachelorarbeit (B.Sc. Computational Engineering 2010) Bachelor's thesis	15 ECTS
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	Modulverantwortliche/r	
5	Inhalt	keine Inhaltsbeschreibung hinterlegt!
6	Lernziele und Kompetenzen	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	keine Einpassung in Studienverlaufsplan hinterlegt!
9	Verwendbarkeit des Moduls	Pflichtmodul Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	schriftlich (5 Monate) mündlich (30 Minuten)
11	Berechnung der Modulnote	schriftlich (80%) mündlich (20%)
12	Turnus des Angebots	keine Angaben zum Turnus des Angebots hinterlegt!
13	Wiederholung der Prüfungen	Die Prüfungen dieses Moduls können nur einmal wiederholt werden.
14	Arbeitsaufwand in Zeitstunden	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
15	Dauer des Moduls	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
16	Unterrichts- und Prüfungssprache	Deutsch
17	Literaturhinweise	

Computer Science

1	Modulbezeichnung 43370	Simulation und Wissenschaftliches Rechnen 1 Simulation and scientific computing 1	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Christoph Pflaum Prof. Dr. Ulrich Rüde	
5	Inhalt	<ul style="list-style-type: none"> • Performance Optimierung für numerische Algorithmen • OpenMP Parallelisierung • Finite Differenzen Diskretisierung im Ort • Praktische Abschätzung des Diskretisierungsfehlers und der Konvergenzgeschwindigkeit numerischer Verfahren • Software Entwicklung im Bereich des wissenschaftlichen Rechnens • MPI Parallelisierung • Finite Differenzen Diskretisierung für zeitabhängige Probleme 	
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • lernen Techniken zur Optimierung von Algorithmen im Bereich des wissenschaftlichen Rechnens • lernen selbständig Algorithmen auf Parallelrechnern zu implementieren und zu optimieren • lernen theoretisch die Stabilität von numerischen Algorithmen zu untersuchen 	
7	Voraussetzungen für die Teilnahme	Voraussetzung ist ein Modul im Bereich Numerik	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	Computer Science Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Übungsleistung Klausur (90 Minuten)	
11	Berechnung der Modulnote	Übungsleistung (bestanden/nicht bestanden) Klausur (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Deutsch Deutsch oder Englisch	
16	Literaturhinweise	<ul style="list-style-type: none"> • Lehrbuch: G. Hager und G. Wellein, Introduction to High Performance Computing for Scientists and Engineers, CRC Press, 2010. 	

- Lehrbuch: Goedecker und Adolfo Hoesle. Performance Optimization of Numerically Intensive Codes, SIAM, 2001.
- Lehrbuch: Gropp, Lusk, Skjellum, Using MPI. The MIT Press, 1999.
- Lehrbuch: Alexandrescu, Modern C++ Design, Generic Programming and Design Patterns. Addison-Wesley, 2001.
- Lehrbuch: Burden, Faires, Numerical Analysis, Brooks, 2001.
- Lehrbuch: Chandra et. al., Programming in OpenMP, Academic Press, 2001.

1	Modulbezeichnung 43610	Computational Engineering 1	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Dietmar Fey	
5	Inhalt	<p>Basiskomponenten eines Rechners</p> <ul style="list-style-type: none"> • Grundlagen der Architektur von Hochleistungsprozessoren (GPGPU, homogene und heterogene Multi-/Vielkern-Prozessoren) • Parallelrechnerarchitekturen • Parallelisierungsstrategien und deren Abbildung auf Architekturen • Leistungsmaße für parallele Architekturen • Einführung Grid-/Cloud-Computing 	
6	Lernziele und Kompetenzen	<p>Fachkompetenz Wissen Lernende können Wissen abrufen und wiedergeben. Sie kennen konkrete Einzelheiten wie Begriffe, Definitionen, Fakten, Daten, Regeln, Gesetzmäßigkeiten, Theorien, Merkmale, Kriterien, Abläufe etc</p> <p>Verstehen Lernende können Beispiele anführen, Aufgabenstellungen interpretieren oder ein Problem in eigenen Worten wiedergeben.</p> <p>Anwenden Lernende können ein neues Problem durch Transfer des Wissens lösen.</p> <p>Analysieren Lernende können ein Problem in einzelne Teile zerlegen und so die Struktur des Problems verstehen; sie können Widersprüche aufdecken, Zusammenhänge erkennen und Folgerungen ableiten und zwischen Fakten und Interpretationen unterscheiden.</p> <p>Lern- bzw. Methodenkompetenz Fähigkeit und Bereitschaft zur Anwendung bestimmter Lern- und Arbeitsmethoden, die zur Entwicklung der anderen Kompetenzen, insbesondere der Fachkompetenz nötig sind.</p> <p>Selbstkompetenz Fähigkeit und Bereitschaft, sich weiterzuentwickeln und das eigene Leben eigenständig und verantwortlich im jeweiligen sozialen, kulturellen bzw. beruflichen Kontext zu gestalten.</p> <p>Sozialkompetenz Fähigkeit und Bereitschaft, zielorientiert mit anderen zusammenzuarbeiten, ihre Interessen und sozialen Situationen zu erfassen, sich mit ihnen rational und verantwortungsbewusst auseinanderzusetzen und zu verständigen sowie die Arbeits- und Lebenswelt mitzugestalten.</p>	

7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 1
9	Verwendbarkeit des Moduls	Computer Science Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Übungsleistung
11	Berechnung der Modulnote	Klausur (100%) Übungsleistung (bestanden/nicht bestanden)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 43871	Simulation und Wissenschaftliches Rechnen 2	7,5 ECTS
2	Lehrveranstaltungen	Übung: Rechnerübung zu Simulation und wissenschaftliches Rechnen 2 Tutorium: Tutorium zu Simulation und Wissenschaftliches Rechnen 2 Übung: Übung zu Simulation und Wissenschaftliches Rechnen 2 Vorlesung: Vorlesung zu Simulation und wissenschaftliches Rechnen 2	- - - -
3	Lehrende	Benjamin Mann Prof. Dr. Christoph Pflaum	

4	Modulverantwortliche/r	Prof. Dr. Christoph Pflaum	
5	Inhalt	<ul style="list-style-type: none"> • Einführung in Mehrgitterverfahren • Theorie und Anwendung der Methode der finiten Elemente • Implementierung von Finite Elemente Verfahren • allgemeine 3-dimensionale Diskretisierungsgitter • Fluidodynamik, Finite Differenzen und Lattice Boltzmann Verfahren • Finite Elemente in der Strukturmechanik • Numerische Lösung der Maxwell'schen Gleichungen 	
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • lernen verschiedene numerische Verfahren zum Lösen partieller Differentialgleichungen kennen • lernen grundlegende Kenntnisse zur Implementierung der entsprechenden Algorithmen • werden in die Entwicklung von Simulationstechniken im Bereich des wissenschaftlichen Rechnens, die • Analyse und Entwicklung von Diskretisierungen für partielle Differentialgleichungen • und die Entwicklung von Software im Bereich des wissenschaftlichen Rechnens eingeführt. 	
7	Voraussetzungen für die Teilnahme	Solides Hintergrundwissen in Ingenieurmathematik und einer höheren Programmiersprache (vorzugsweise C/C++)	
8	Einpassung in Studienverlaufsplan	Semester: 6	
9	Verwendbarkeit des Moduls	Computer Science Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Übungsleistung	
11	Berechnung der Modulnote	Klausur (100%) Übungsleistung (bestanden/nicht bestanden)	
12	Turnus des Angebots	nur im Sommersemester	

13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> • Briggs, Henson, McCormick, A Multigrid Tutorial. SIAM, ISBN 0-89871-462-1. • Strang, Fix, An Analysis of the Finite Element Method. Wellesley-Cambridge Press, ISBN 0-9614088-8-X. • Axelsson, Barker, Finite Element Solution of Boundary Value Problems. Siam, ISBN 0-89871-499-0. • Braess, Finite Elemente. Springer, ISBN 3-540-61905-4. • Braess, Finite elements. Cambridge University Press, ISBN 0521011957. • Großmann, Roos, Numerik partieller Differentialgleichungen. Teubner, ISBN 3-519-02089-0. • Großmann, Roos, Numerische Behandlung partieller Differentialgleichungen. Teubner, ISBN 3-519-22089-X. • Grossmann, Roos, Stynes, Numerical treatment of partial differential equations. Springer, ISBN 978-3-540-71582-5.

1	Modulbezeichnung 93050	Algorithmen und Datenstrukturen Algorithms and data structures	10 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Michael Philippsen	
5	Inhalt	<ul style="list-style-type: none"> • Grundlagen der Programmierung • Datenstrukturen • Objektorientierung • JAVA-Grundkenntnisse • Aufwandsabschätzungen • Grundlegende Algorithmen 	
6	Lernziele und Kompetenzen	<p><u>A - Fachkompetenz:</u> Die Studierenden...</p> <p>1.) Grundlagen der Programmierung in Java</p> <ul style="list-style-type: none"> • interpretieren Syntaxdiagramme für grundlegende Programmstrukturen und übertragen diese in entsprechenden Java-Code • deklarieren und verwenden Variablen mit adäquatem Java-Datentyp (primitive Typen, Reihungen, Zeichenketten) • überprüfen die Zulässigkeit der Variablendeklaration und -Wertzuzuweisung nach Java-Typ-Regeln • bestimmen den Datentyp und den Wert eines Java-Ausdrucks mit primitivem Datentyp und zugehörigen Operatoren • überführen einfache mathematische Ausdrücke in Java-Code • werten zusammengesetzte Bedingungen nach den Regeln der strikten bzw. faulen Auswertung für Java aus • konzipieren zu einer gegebenen Aufgabenstellung einen Algorithmus • implementieren einfache Algorithmen in Java unter Verwendung verschiedener Kontrollstrukturen • bestimmen die Gültigkeitsbereiche der Variablen anhand der Blockstruktur eines Java-Programms • strukturieren Java-Code in Methoden und entwickeln wiederverwendbare Funktionen <p>2.) Rekursion</p> <ul style="list-style-type: none"> • beurteilen den Typ der Rekursion für gegebene Java-Methoden • entwerfen rekursive Algorithmen zu einer gegebenen Problemstellung unter Anwendung des Induktionsprinzips, des Teile-und-Herrsche-Prinzips sowie des Rücksetzverfahrens und implementieren diese jeweils in Java • entwickeln effizientere Lösungen, indem sie rekursive Methoden in endrekursive bzw. iterative Methoden 	

umwandeln, implementieren diese jeweils in Java-Code und bewerten deren Laufzeit- und Speicherverbrauch

- bewerten und verbessern rekursive Lösungen unter Verwendung von Dynamischer Programmierung und implementieren diese in Java-Code

3.) Aufwandsanalyse

- analysieren den Laufzeitaufwand und den Speicherbedarf verschiedener Implementierungen
- klassifizieren den asymptotischen Laufzeitaufwand anhand der Komplexitätsklassen des O-Kalküls
- unterscheiden verschiedene Sortierverfahren (Blasensortierung, Sortieren durch Auswählen/Einfügen, Haldensortierung, Sortieren durch Verschmelzen/Zerlegen/Fachverteilen) hinsichtlich ihres Laufzeit- und Speicherplatzbedarfs

4.) Objekt-Orientierte Programmierung in Java

- implementieren Java-Klassen gemäß textueller oder graphischer (UML) Spezifikation
- wenden Verfahren zur systematischen Ableitung von Klassen und Attributen (Hauptwortextraktion), ihren statischen Beziehungen (Vererbung, Polymorphie, Assoziationen) und ihrem dynamischen Zusammenspiel (CRC, Kollaboration) aus einer textuellen Problemstellung an und entwickeln so kleine objekt-orientierte Java-Programme
- instantiiieren Klassen und verwenden Objektvariablen sachgerecht
- unterscheiden statische und dynamische Bindung gemäß Polymorphie-Konzept von Java und wenden die Erkenntnisse sachgerecht bei der Entwicklung eigener Applikationen an

5.) Robustes Programmieren

- wenden Checklisten an, um typische Programmierfehler im Vorfeld zu vermeiden oder nach der Programmierung zu identifizieren
- benutzen verschiedenen Möglichkeiten zur Absicherung gegen Fehlersituationen und zur Fehlerrückmeldung (Rückgabewert, Ausnahmebehandlung)
- wenden Junit zum Testen von Java-Programmen an
- setzen Verfahren und Werkzeuge zur systematischen Lokalisierung und Behebung von Programmfehlern an (Debugging) und verbessern ihre Lösungen auf diese Weise iterativ

6.) Elementare Datentypen

- übertragen eine Spezifikation in Form eines Abstrakten Datentyps (ADT) in ein gleichwertiges Java-Modul
- erstellen eine formale Spezifikation eines Datentyps in Form eines Abstrakten Datentyps (ADT) aus einer gegebenen textuellen Beschreibung
- verstehen die grundlegende Behälterdatentypen (Liste, Stapel, Schlange, Streutabelle) und deren Eigenschaften

		<p>(insbesondere Laufzeit- und Speicherplatzbedarf ihrer Operationen)</p> <ul style="list-style-type: none"> • verwenden generische Behälterdatentypen sachgerecht in eigenen Programmen • kennen die Verfügbarkeit generischer Behälterdatentypen in der Java-API und erschließen sich bei Bedarf selbst neue Datentypen sowie deren Funktionen aus der zugehörigen API-Spezifikation für die Verwendung in eigenen Programmen <p>7.) Bäume und Graphen</p> <ul style="list-style-type: none"> • bewerten verschiedene Baum- und Graphdarstellungen hinsichtlich Zeitaufwand und Speicherbedarf • unterscheiden und klassifizieren die grundlegenden Baum-Arten (Suchbaum, AVL-Baum, Halde) • wenden die Grundoperationen (Einfügen, Suchen, Löschen, ggf. Restrukturieren) anhand von Beispieldaten auf gegebene Bäume artgerecht an • implementieren und verwenden verschiedene Baumstrukturen sachgerecht in eigenen Java-Programmen • führen verschiedene Durchlaufmöglichkeiten (Tiefensuche (DFS), Breitensuche (BFS)) für Graphen und Bäume auf Beispieldaten aus und setzen diese zielführend in eigenen Java-Programmen ein • wenden grundlegende Graphalgorithmen (Dijkstra, Floyd, Prim, Kruskal) auf Beispieldaten an und implementieren diese Verfahren in Java-Code <p><u>B - Selbst- und Sozialkompetenz:</u> Die Studierenden...</p> <ul style="list-style-type: none"> • organisieren sich selbständig zu Gruppen und koordinieren in gegenseitiger Absprache den organisatorischen und technischen Ablauf der Gruppenarbeiten • kommunizieren und erarbeiten gemeinsam Lösungen für theoretische Fragestellungen und praktische Programmieraufgaben in Rahmen von Gruppenaufgaben • planen und wenden zielgerichtet Maßnahmen zu gegenseitigen Qualitätssicherung der eingereichten Lösungen an (prüfen wechselseitig die Gruppenabgaben) • verantworten gemeinsam das Ergebnis ihrer Gruppenarbeit, deren Bewertung für beide Gruppenpartner gleichermaßen gilt
7	Voraussetzungen für die Teilnahme	<p>Wichtiger Hinweis: Die Vorlesung fand letztmals im Wintersemester 2021/22 statt. Übungsbetrieb und Klausur werden vorerst noch in jedem Semester angeboten, allerdings endet das Angebot in naher Zukunft.</p>
8	Einpassung in Studienverlaufsplan	Semester: 1
9	Verwendbarkeit des Moduls	Computer Science Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (120 Minuten) Übungsleistung

11	Berechnung der Modulnote	Klausur (100%) Übungsleistung (bestanden/nicht bestanden)
12	Turnus des Angebots	Unregelmäßig
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 120 h Eigenstudium: 180 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Lehrbuch: Saake, Sattler: Algorithmen und Datenstrukturen - Eine Einführung mit JAVA

Übung: SP1-T07 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T08 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T09 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T11 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T12 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T14 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T16 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T18 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T21 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-Tutorenbesprechung (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T02 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T04 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T06 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T10 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T13 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T15 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T17 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T19 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP1-T20 (2 SWS, SoSe 2025)	2,5 ECTS
Vorlesung: Systemprogrammierung 1 (2 SWS, SoSe 2025)	2,5 ECTS
Übung: SP2-Ü T01 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T02 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T03 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T04 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T05 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T06 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T07 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T08 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T09 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T11 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T12 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü T18 (2 SWS, WiSe 2025)	-
Übung: SP2-Ü Ta, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter	-

		<p>https://sys.cs.fau.de/lehre/current/sp2/uebung#termine (2 SWS, WiSe 2025)</p> <p>Übung: SP2-Ü Tb, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter https://sys.cs.fau.de/lehre/current/sp2/uebung#termine) (2 SWS, WiSe 2025)</p> <p>Übung: SP2-Ü Ti, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter https://sys.cs.fau.de/lehre/current/sp2/uebung#termine) (2 SWS, WiSe 2025)</p> <p>Übung: SP2-Ü Tk, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter https://sys.cs.fau.de/lehre/current/sp2/uebung#termine) (2 SWS, WiSe 2025)</p> <p>Übung: SP2-Ü Tm, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter https://sys.cs.fau.de/lehre/current/sp2/uebung#termine) (2 SWS, WiSe 2025)</p> <p>Übung: SP2-Ü Tn, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter https://sys.cs.fau.de/lehre/current/sp2/uebung#termine) (2 SWS, WiSe 2025)</p> <p>Übung: SP2-Ü To, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter https://sys.cs.fau.de/lehre/current/sp2/uebung#termine) (2 SWS, WiSe 2025)</p> <p>Übung: SP2-Ü Ts, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter https://sys.cs.fau.de/lehre/current/sp2/uebung#termine) (2 SWS, WiSe 2025)</p> <p>Übung: SP2-Ü Tu, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter https://sys.cs.fau.de/lehre/current/sp2/uebung#termine) (2 SWS, WiSe 2025)</p> <p>Übung: SP2-Ü Tv, eventuell nicht besetzt (alle tatsächlich stattfindenden Termine finden sie unter https://sys.cs.fau.de/lehre/current/sp2/uebung#termine) (2 SWS, WiSe 2025)</p>	- - - - - - - - -
3	Lehrende	<p>Thomas Preisner Luis Gerhorst Dr.-Ing. Jürgen Kleinöder Prof. Dr.-Ing. Rüdiger Kapitza</p>	

4	Modulverantwortliche/r	Dr.-Ing. Jürgen Kleinöder
---	-------------------------------	---------------------------

		Prof. Dr.-Ing. Wolfgang Schröder-Preikschat
5	Inhalt	<ul style="list-style-type: none"> • Grundlagen von Betriebssystemen (Adressräume, Speicher, Dateien, Prozesse, Koordinationsmittel; Betriebsarten, Einplanung, Einlastung, Virtualisierung, Nebenläufigkeit, Koordination/Synchronisation) • Abstraktionen/Funktionen UNIX-ähnlicher Betriebssysteme • Programmierung von Systemsoftware • C, Make, UNIX-Shell (Solaris, Linux, MacOS X)
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • erwerben fundierte Kenntnisse über Grundlagen von Betriebssystemen • verstehen Zusammenhänge, die die Ausführungen von Programmen in vielschichtig organisierten Rechensystemen ermöglichen • erkennen Gemeinsamkeiten und Unterschiede zwischen realen und abstrakten (virtuellen) Maschinen • erlernen die Programmiersprache C • entwickeln Systemprogramme auf Basis der Systemaufrufchnittstelle UNIX-ähnlicher Betriebssysteme
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 2;3
9	Verwendbarkeit des Moduls	Computer Science Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur mit MultipleChoice (120 Minuten)
11	Berechnung der Modulnote	Klausur mit MultipleChoice (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 180 h Eigenstudium: 120 h
14	Dauer des Moduls	2 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • Lehrbuch: Betriebssysteme Grundlagen, Entwurf, Implementierung, Wolfgang Schröder-Preikschat, 2008

1	Modulbezeichnung 97090	Simulation und Modellierung I Simulation and modelling I	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Reinhard German	
5	Inhalt	<p>Das Modul vermittelt die Grundlagen der diskreten Ereignissimulation und beinhaltet</p> <ul style="list-style-type: none"> • diskrete Simulation • analytische Modellierung (z.B. Warteschlangen) • Eingabemodellierung (z.B. Fitting-Verfahren) • Zufallszahlenerzeugung • statistische Ausgabeanalyse • Modellierungsparadigmen (u.a. Ereignis-/Prozessorientierung, Warteschlangen, Automaten, Petri-Netze, UML, graphische Bausteine) • kontinuierliche und hybride Simulation • Simulationssoftware • Fallstudien <p>Content: Overview of the various kinds of simulation</p> <ul style="list-style-type: none"> • discrete simulation (computational concepts, simulation of queuing systems, simulation in Java, professional simulation tools) • required probability concepts and statistics, modeling paradigms (e.g., event/process oriented, queuing systems, Petri nets, UML statecharts) • input modeling (selecting input probability distributions) • random number generation (linear congruential generators and variants, generating random variates) • output analysis (warm-up period detection, independent replications, result presentation) • continuous and hybrid simulation (differential equations, numerical solution, hybrid statecharts) • simulation software, case studies, parallel and distributed simulation. 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • erwerben Kenntnisse über Verfahren und Realisierungsmöglichkeiten der diskreten Simulation mit Ausblick auf andere Simulationsarten • erwerben Kenntnisse über statistische Aspekte der Simulation, die für die Anwendung wichtig sind • wenden statistische Methoden zur Analyse und Bewertung von Eingabe- sowie Ausgabedaten an 	

		<ul style="list-style-type: none"> • erwerben praktische Erfahrung mit kommerziellen Simulationswerkzeugen • erwerben Erfahrungen bei der Simulation in verschiedenen Anwendungsbereichen (u.a. Rechnernetze, Fertigungssysteme, Materialflusssysteme) • entwickeln eigenständig anhand von Beispielaufgaben Simulationsmodelle unter Verwendung verschiedener Modellierungsparadigmen • können in Gruppen kooperativ und verantwortlich arbeiten <p>Learning targets and competences: Students</p> <ul style="list-style-type: none"> • gain knowledge about methods and realization possibilities of discrete simulation with an outlook on other types of simulation • gain knowledge of statistical aspects of simulation that are important for practice • apply statistical methods for analysis and evaluation of input and output data • gain hands-on experience with commercial simulation tools • gain experience in simulation in various fields of application (including computer networks, manufacturing systems, material flow systems) • independently develop simulation models on the basis of sample tasks using different modeling paradigms • can work in groups cooperatively and responsibly
7	Voraussetzungen für die Teilnahme	<p>elementare Programmierkenntnisse, vorzugsweise in Java, Mathematikkennnisse in Analysis, wie z.B. im 1. Semester der angewandten Mathematik vermittelt</p> <p>Recommended background knowledge: basic programming skills, preferably in Java, mathematics skills in analysis, such as taught in the first semester in applied mathematics.</p>
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Computer Science Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>Klausur (90 Minuten) Prüfungsleistung/examination: Klausur, benotet, 5 ETCS/written exam, graded, 5 ETCS Dauer (in Minuten)/duration (in minutes): 90 Anteil an der Berechnung der Modulnote/Share in the calculation of the module grade: 100.0 % Die im Rahmen der Übung gestellten (zwei-)wöchentlichen Übungsaufgaben müssen bestanden werden, um das Gesamtmodul anrechnen lassen zu können. Die Übung gilt als bestanden, wenn mindestens 50% der Punkte korrekt bearbeitet wurden. Die Bearbeitung erfolgt in Gruppen von 3 oder 4 Studenten. Die Abgabe erfolgt in Präsenz zu dedizierten Übungsterminen. Wurden mindestens 70% der Punkte erreicht, wird die Endnote der bestandenen schriftlichen Prüfung entsprechend einer Notenstufe (0.3 oder 0.4) verbessert.</p>

		<p>Wurden mindestens 90% der Punkte erreicht, wird die Endnote der bestandenen schriftlichen Prüfung entsprechend zwei Notenstufe (0.6 oder 0.7) verbessert.</p> <p>-----</p> <p>The (bi-)weekly exercise tasks must be passed in order to receive credit for the entire module. The exercise is considered to be passed if at least 50% of the points have been correctly processed. The work is done in groups of 3 or 4 students. The submission is done in presence on dedicated exercise dates.</p> <p>If at least 70% of the points are achieved, the grade of the passed written exam will be improved by one grade level (0.3 or 0.4).</p> <p>If at least 90% of the points are achieved, the grade of the passed written exam will be improved by two grade levels (0.6 or 0.7).</p>
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	Law, "Simulation Modeling and Analysis, 5th ed., McGraw Hill, 2014

Mathematics

1	Modulbezeichnung 64620	Numerik I für Ingenieure Numerics for engineers I	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Dr. Michael Fried apl. Prof. Dr. Wilhelm Merz	
5	Inhalt	<ul style="list-style-type: none"> • Elementare Numerik: Direkte und iterative Lösungsverfahren bei linearen Gleichungssystemen, Interpolation mit Newton-Polynomen und Splines, Quadratur mit Newton-Côtes-Formeln, Extrapolation nach Romberg • Numerik gewöhnlicher Differentialgleichungen: Verschiedene Runge-Kutta Methoden als Einschrittverfahren, Konsistenz, Stabilität- und Konvergenzaussage, Mehrschrittverfahren 	
6	Lernziele und Kompetenzen	<p>Die Studierenden lernen</p> <ul style="list-style-type: none"> • verschiedene numerische Methoden zur Lösung linearer Gleichungssysteme • verschiedene Methoden zu beurteilen • Interpolationstechniken und Güte der Approximation • grundlegende Quadraturverfahren und die Beurteilung solcher • grundlegende Diskretisierungsmethoden bei gewöhnlichen Differentialgleichungen • Beurteilung dieser Methoden und Verfahren • algorithmische Umsetzung o.g. Verfahren als Grundlage für Computer-Codes 	
7	Voraussetzungen für die Teilnahme	Kurse Mathematik für Ingenieure I, II und III	
8	Einpassung in Studienverlaufsplan	Semester: 3	
9	Verwendbarkeit des Moduls	Mathematics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (60 Minuten)	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Deutsch Englisch	
16	Literaturhinweise	<ul style="list-style-type: none"> • Skripte des Dozenten 	

1	Modulbezeichnung 64631	Numerik II für Ingenieure Numerics for engineers II	5 ECTS
2	Lehrveranstaltungen	Übung: Num2U (2 SWS) Vorlesung: Numerik II für Ingenieure (2 SWS)	- -
3	Lehrende	Dr. Michael Fried	

4	Modulverantwortliche/r	apl. Prof. Dr. Wilhelm Merz	
5	Inhalt	*Numerik partieller Differentialgleichungen* Finite Differenzenmethode, Stabilität, Konsistenz, Konvergenz, Einführung finite Elementmethode bei elliptischen Problemen, Fehlerschätzer	
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • erklären verschiedene Diskretisierungsmethoden • beurteilen diese Diskretisierungsmethoden • leiten Finite Elemente Diskretisierungen elliptischer Probleme her • folgern Aussagen anhand grundlegender Beweistechniken aus oben genannten Bereichen • konstruieren Algorithmen zu Finite Elemente Diskretisierungen • erklären Fehlerschätzer 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	Mathematics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (60 Minuten)	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Sommersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Deutsch	
16	Literaturhinweise	Skripte des Dozenten H. Jung, M. Langer, Methode der Finiten Elemente, Teubner P. Knabner, L. Angermann, Numerik partieller Differentialgleichungen, Springer	

1	Modulbezeichnung 67520	Mathematik für CE 1 Mathematics for CE 1	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Dr. Michael Fried PD Dr. Cornelia Schneider	
5	Inhalt	<p>*Grundlagen*</p> <p>Aussagenlogik, Mengen, Relationen, Abbildungen</p> <p>*Zahlensysteme*</p> <p>natürliche, ganze, rationale und reelle Zahlen, komplexe Zahlen</p> <p>*Vektorräume*</p> <p>Grundlagen, Lineare Abhängigkeit, Spann, Basis, Dimension, euklidische Vektor- und Untervektorräume, affine Räume</p> <p>*Matrizen, Lineare Abbildungen, Lineare Gleichungssysteme*</p> <p>Matrixalgebra, Lösungsstruktur linearer Gleichungssysteme, Gauß-Algorithmus, inverse Matrizen, Matrixtypen, lineare Abbildungen, Determinanten, Kern und Bild, Eigenwerte und Eigenvektoren, Basis, Ausgleichsrechnung</p> <p>*Grundlagen Analysis einer Veränderlichen*</p> <p>Grenzwert, Stetigkeit, elementare Funktionen, Umkehrfunktionen</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • verstehen grundlegende Begriffe und Strukturen der Mathematik • erklären den Aufbau von Zahlensystemen im Allgemeinen und der Obengenannten im Speziellen • rechnen mit komplexen Zahlen in Normal- und Polardarstellung und Wechseln zwischen diesen Darstellungen • berechnen lineare Abhängigkeiten, Unterräume, Basen, Skalarprodukte, Determinanten • vergleichen Lösungsmethoden zu linearen Gleichungssystemen • bestimmen Lösungen zu Eigenwertproblemen • überprüfen Eigenschaften linearer Abbildungen und Matrizen • überprüfen die Konvergenz von Zahlenfolgen •ermitteln Grenzwerte und überprüfen Stetigkeit •entwickeln Beweise anhand grundlegender Beweismethoden aus den genannten Themenbereichen •kennen eine regelmäßige selbstständige Nachbereitung und Anwendung des Vorlesungsstoffes 	
7	Voraussetzungen für die Teilnahme	Keine	

8	Einpassung in Studienverlaufsplan	Semester: 1
9	Verwendbarkeit des Moduls	Mathematics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Übungsleistung
11	Berechnung der Modulnote	Klausur (100%) Übungsleistung (bestanden/nicht bestanden)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Empfohlene Begleitlektüre: Skripte des Dozenten M. Fried: Mathematik für Ingenieure I für Dummies. Wiley A. Hoffmann, B. Marx, W. Vogt: Mathematik für Ingenieure 1. Pearson v. Finckenstein et.al: Arbeitsbuch Mathematik fuer Ingenieure: Band I Analysis und Lineare Algebra. Teubner-Verlag 2006, ISBN 9783835100343

1	Modulbezeichnung 67530	Mathematik für CE 2 Mathematics for CE 2	10 ECTS
2	Lehrveranstaltungen	Übung: IngMathA2U (2 SWS) Übung: Übungen zur Mathematik für Ingenieure A2 : CE, EEI, BP-E, MT (2 SWS) Übung: Übungen zur Mathematik für Ingenieure A2 : CE, EEI, BP-E, MT diese Gruppe findet voraussichtlich nicht statt! (2 SWS) Vorlesung: Mathematik für Ingenieure A2 : CE, EEI, BP-E, MT (6 SWS)	- - - -
3	Lehrende	PD Dr. Cornelia Schneider	

4	Modulverantwortliche/r	Dr. Michael Fried	
5	Inhalt	<p>*Differentialrechnung einer Veränderlichen*</p> <p>Ableitung mit Rechenregeln, Mittelwertsätze, L'Hospital, Taylor-Formel, Kurvendiskussion</p> <p>*Integralrechnung einer Veränderlichen*</p> <p>Riemann-Integral, Hauptsatz der Infinitesimalrechnung, Mittelwertsätze, Partialbruchzerlegung, uneigentliche Integration</p> <p>*Folgen und Reihen*</p> <p>reelle und komplexe Zahlenfolgen, Konvergenzbegriff und -sätze, Folgen und Reihen von Funktionen, gleichmäßige Konvergenz, Potenzreihen, iterative Lösung nichtlinearer Gleichungen</p> <p>*Grundlagen Analysis mehrerer Veränderlicher*</p> <p>Grenzwert, Stetigkeit, Differentiation, partielle Ableitungen, totale Ableitung, allgemeine Taylor-Formel, Extremwertaufgaben, Extremwertaufgaben mit Nebenbedingungen, Theorem über implizite Funktionen</p> <p>*Gewöhnliche Differentialgleichungen*</p> <p>Explizite Lösungsmethoden, Existenz- und Eindeutungssätze, Lineare Differentialgleichungen, Systeme von Differentialgleichungen, Eigen- und Hauptwertaufgaben, Fundamentalsysteme, Stabilität</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • analysieren Funktionen einer reellen Veränderlichen mit Hilfe der Differentialrechnung • berechnen Integrale von Funktionen mit einer reellen Veränderlichen • stellen technisch-naturwissenschaftliche Problemstellungen mit mathematischen Modellen dar und lösen diese • erklären den Konvergenzbegriff bei Folgen und Reihen • berechnen Grenzwerte und rechnen mit diesen • analysieren und klassifizieren Funktionen mehrerer reeller Veränderlicher an Hand grundlegender Eigenschaften • wenden grundlegende Beweistechniken in o.g. Bereichen an 	

		<ul style="list-style-type: none"> • klassifizieren gewöhnliche Differentialgleichungen nach Typen • wenden elementare Lösungsmethoden auf Anfangswertprobleme bei gewöhnlichen Differentialgleichungen an • wenden allgemeine Existenz- und Eindeutigkeitsresultate an • erschließen den Zusammenhang zwischen Analysis und linearer Algebra • wenden die erlernten mathematischen Methoden auf die Ingenieurwissenschaften an • erkennen die Vorzüge einer regelmäßigen Nachbereitung und Vertiefung des Vorlesungsstoffs
7	Voraussetzungen für die Teilnahme	Besuch der Vorlesung Mathematik für Ingenieure I
8	Einpassung in Studienverlaufsplan	Semester: 2
9	Verwendbarkeit des Moduls	Mathematics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Übungsleistung Klausur (120 Minuten)
11	Berechnung der Modulnote	Übungsleistung (bestanden/nicht bestanden) Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 120 h Eigenstudium: 180 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>v. Finckenstein et.al: Arbeitsbuch Mathematik fuer Ingenieure: Band I Analysis und Lineare Algebra. Teubner-Verlag 2006, ISBN 9783835100343</p> <p>M. Fried: Mathematik für Ingenieure I für Dummies. Wiley</p> <p>M. Fried: Mathematik für Ingenieure II für Dummies. Wiley</p> <p>A. Hoffmann, B. Marx, W. Vogt: Mathematik für Ingenieure 1, 2. Pearson</p> <p>H. Heuser: Gewöhnliche Differentialgleichungen. Teubner</p> <p>W. Merz, P. Knabner: Mathematik für Ingenieure und Naturwissenschaftler, Springer, 2013</p>

1	Modulbezeichnung 67540	Mathematik für CE 3 Mathematics for CE 3	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Dr. Michael Fried	
5	Inhalt	<p>*Funktionentheorie:*</p> <p>Elementare Funktionen komplexer Variablen, holomorphe Funktionen, Integralsatz von Cauchy, Residuentheorie</p> <p>*Vektoranalysis*</p> <p>Potentiale, Volumen-, Oberflächen- und Kurvenintegrale, Parametrisierung, Transformationssatz, Integralsätze, Differentialoperatoren</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • analysieren elementare komplexe Funktionen • überprüfen und beurteilen Eigenschaften dieser Funktionen • wenden den Integralsatz von Cauchy an • wenden die Residuentheorie an • berechnen Integrale über mehrdimensionale Bereiche • beobachten Zusammenhänge zwischen Volumen-, Oberflächen- und Kurvenintegralen • ermitteln Volumen-, Oberflächen- und Kurvenintegrale • wenden grundlegende Differentialoperatoren an. • folgern Aussagen anhand grundlegender Beweistechniken in o.g. Bereichen • beachten die Vorzüge einer regelmäßigen Nachbereitung und Vertiefung des Vorlesungsstoffes 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 3	
9	Verwendbarkeit des Moduls	Mathematics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (60 Minuten) Übungsleistung	
11	Berechnung der Modulnote	Klausur (100%) Übungsleistung (bestanden/nicht bestanden)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h	
14	Dauer des Moduls	1 Semester	

15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>Empfohlene Begleitlektüre:</p> <p>Skripte des Dozenten</p> <p>M. Fried: Mathematik für Ingenieure II für Dummies. Wiley</p> <p>A. Hoffmann, B. Marx, W. Vogt: Mathematik für Ingenieure 1, 2. Pearson</p> <p>v. Finckenstein et.al: Arbeitsbuch Mathematik fuer Ingenieure: Band I und II. Vieweg+Teubner</p>

1	Modulbezeichnung 67550	Mathematik für CE 4 Mathematics for CE 4	5 ECTS
2	Lehrveranstaltungen	Übung: IngMathA4U (2 SWS) Vorlesung: Mathematik für Ingenieure A4 : EEI,CE,MT (2 SWS)	- -
3	Lehrende	Dr. Michael Fried	

4	Modulverantwortliche/r	Dr. Michael Fried
5	Inhalt	*Kontinuierliche Wahrscheinlichkeitsrechnung* Ereignisraum, Wahrscheinlichkeitsraum, stetige Zufallsvariable, Wahrscheinlichkeitsdichte, Verteilungsfunktion, charakteristische Größen *Stochastische Prozesse* Orthogonalität, Unkorreliertheit, weißes Rauschen, Gauß-Prozesse, Stationarität, Ergodizität, Leistungsdichtespektrum, lineare Systeme, Zufallsprozesse
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • untersuchen oben genannte grundlegende Begriffe und Methoden der Stochastik • berechnen obige charakteristische Größen und Erwartungswerte • untersuchen oben genannte grundlegende Begriffe und Methoden für stochastische Prozesse • berechnen obige charakteristische Größen und Erwartungswerte für stochastische Prozesse • schätzen die Vorzüge einer regelmäßigen Nachbereitung und Vertiefung des Vorlesungsstoffes
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Mathematics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Übungsleistung Klausur (60 Minuten)
11	Berechnung der Modulnote	Übungsleistung (bestanden/nicht bestanden) Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Skripte des Dozenten

A. Hoffmann, B. Marx, W. Vogt, Mathematik für Ingenieure 1,2, Pearson

K. Finck von Finckenstein, J. Lehn et. al., Arbeitsbuch für Ingenieure,
Band I und II, Teubner

R.G. Brown, P.Y.C. Hwang, Introduction to Random Signals and Applied
Kalman Filtering, John Wiley & Sons

NF Automatic Control

1	Modulbezeichnung 92241	Modeling of Control Systems	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Thomas Moor	
5	Inhalt	<ul style="list-style-type: none"> • Ordinary differential equations as models of engineering processes • State space representation and linearisation • Control engineering models of mechanical systems • Control engineering models of chemical processes • Numerical methods for the solution of ordinary differential equations 	
6	Lernziele und Kompetenzen	<p>The students</p> <ul style="list-style-type: none"> • explain how to derive models for engineering processes • develop models for the control of basic technical processes • develop models for complex mechanical systems • explain established models for basic chemical processes • discuss and evaluate methods for the numerical solution of ordinary differential equations 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	NF Automatic Control Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Englisch	
16	Literaturhinweise	Woods, R.L., Lawrence, K.L.: Modeling and Simulation of Dynamic Systems, Prentice Hall, 1997	

1	Modulbezeichnung 92440	Regelungstechnisches Praktikum Laboratory: Control Engineering	5 ECTS
2	Lehrveranstaltungen	Praktikum: Regelungstechnisches Praktikum für CE (3 SWS)	2,5 ECTS
3	Lehrende	Daniel Landgraf Prof. Dr. Thomas Moor	

4	Modulverantwortliche/r	Prof. Dr. Thomas Moor	
5	Inhalt	<ul style="list-style-type: none"> • Matlab/Simulink zum Entwurf von Regelkreisen • Modellbildung zur Regelung diverser Laboraufbauten • Reglerentwurf für diverse Laboraufbauten 	
6	Lernziele und Kompetenzen	Die Teilnehmer dieser Veranstaltung <ul style="list-style-type: none"> • wenden die eingebrachten methodischen Grundlagen am Laboraufbau bzw. im Simulationsexperiment an, • hinterfragen ihre Beobachtungen anhand der eingebrachten methodischen Grundlagen, • schildern die verwendete Gerätetechnik und deren prinzipielle Funktionsweise. 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	NF Automatic Control Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Praktikumsleistung	
11	Berechnung der Modulnote	Praktikumsleistung (100%)	
12	Turnus des Angebots	in jedem Semester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Deutsch	
16	Literaturhinweise		

1	Modulbezeichnung 97040	Einführung in die Regelungstechnik Introduction to control engineering	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Thomas Moor	
5	Inhalt	<p>Grundlagen der klassischen Regelungstechnik</p> <ul style="list-style-type: none"> • Lineare zeitinvariante Eingrößensysteme im Frequenz- und Zeitbereich • Sensitivitäten des Standardregelkreises • Bode-Diagramm und Nyquist-Kriterium • Entwurf von Standardreglern • Algebraische Entwurfsmethoden • Erweiterte Regelkreisarchitekturen <p>Anwendungsstudien aus den Bereichen</p> <ul style="list-style-type: none"> • Mechanische Systeme • Verfahrenstechnische Prozesse 	
6	Lernziele und Kompetenzen	<p>Die Teilnehmer</p> <ul style="list-style-type: none"> • erklären und illustrieren die vorgestellten Entwurfsziele und Entwurfsverfahren anhand von Beispielen, • erkennen elementare mathematische Zusammenhänge zwischen Systemtheorie und Reglerentwurf, • können die vorgestellten Entwurfsverfahren auf einfache Anwendungsfälle anwenden und kritisch hinterfragen, • erkennen im Anwendungskontext gegenläufige oder sich ausschließende Entwurfsziele. 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 3	
9	Verwendbarkeit des Moduls	NF Automatic Control Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Deutsch	

16	Literaturhinweise	Unbehauen, H.: Regelungstechnik I, Vieweg, 1982 Glattfelder, A.H., Schaufelberger, W.: Lineare Regelsysteme, VDH Verlag, 1996 Goodwin, G.C., et al.: Control System Design, Prentice Hall, 2001
----	--------------------------	---

1	Modulbezeichnung 97060	Regelungstechnik B (Zustandsraummethoden) Control engineering B (State-space methods)	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Knut Graichen	
5	Inhalt	<p>Das Modul vermittelt die Grundlagen zur Beschreibung und Untersuchung von linearen dynamischen Systemen mit mehreren Ein- und Ausgangsgrößen im Zustandsraum sowie den zustandsraumbasierten Regler- und Beobachterentwurf. Die Inhalte sind:</p> <ul style="list-style-type: none"> • Motivation der Zustandsraumbetrachtung dynamischer Systeme in der Regelungstechnik • Zustandsraumdarstellung dynamischer Systeme und deren Vereinfachung durch Linearisierung • Analyse linearer und zeitinvarianter Systeme: Stabilität, Steuerbarkeit, Beobachtbarkeit, Zusammenhang mit Ein-/Ausgangsbetrachtung • Auslegung von linearen Zustandsreglern für lineare Eingrößensysteme • Erweiterte Regelkreisstrukturen, insbesondere Vorsteuerung und Störgrößenkompensation • Entwurf von Zustands- und Störgrößenbeobachtern und Kombination mit Zustandsreglern (Separationsprinzip) 	
6	Lernziele und Kompetenzen	<p>Die Studierenden können</p> <ul style="list-style-type: none"> • die Vorzüge der Zustandsraumbetrachtung im Vergleich zur Ein-/Ausgangsbetrachtung darlegen. • für dynamische Systeme die Zustandsgleichungen aufstellen und durch Linearisierung vereinfachen. • für LZI-Systeme die Zustandsgleichungen in Normalformen transformieren. • Stabilität, Steuer- und Beobachtbarkeit von Zustandssystemen definieren und LZI-Systeme daraufhin untersuchen. • ausführen, wie diese Eigenschaften mit den Eigenwerten und Nullstellen von LZI-Zustandssystemen zusammenhängen. • den Aufbau einer Zwei-Freiheitsgrade-Zustandsregelung angeben und die Zweckbestimmung ihrer einzelnen Komponenten erläutern. • realisierbare Vorsteuerungen zur Einstellung des Sollverhaltens entwerfen. • Zielstellung und Aufbau eines Zustandsbeobachters erläutern. • diesen zu einem Störbeobachter erweitern und Störaufschaltungen zur Kompensation von Dauerstörungen konzipieren. 	

		<ul style="list-style-type: none"> • beobachterbasierte Zustandsregelungen durch Eigenwertvorgabe entwerfen. • die Vorlesungsinhalte auf verwandte Problemstellungen übertragen und sich die Zustandsraummethoden der Regelungstechnik selbständig weiter erschließen.
7	Voraussetzungen für die Teilnahme	Empfohlene Vorkenntnisse: Vektor- und Matrizenrechnung sowie Grundlagen der Regelungstechnik (klassische Frequenzbereichsmethoden; kann auch parallel gehört werden, siehe Regelungstechnik A)
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	NF Automatic Control Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Die Summe der in den Online-Tests erzielten Punktzahl wird zu max. 10% auf die Klausurpunktzahl angerechnet. Hiermit ist eine Verbesserung der Klausurbewertung um bis zu 0,7 Notenpunkte möglich. Die Anrechnung erfolgt nur, wenn Sie die Prüfung an sich mit der Mindestnote 4,0 bestanden haben. Der Bonus kann nur einmal im Prüfungszeitraum der Vorlesung angerechnet werden, entweder zum Haupttermin nach Vorlesungsende oder zum Nachholtermin im Folgesemester, wenn der Haupttermin nicht wahrgenommen wurde.
11	Berechnung der Modulnote	Klausur (100%) Die Summe der in den Online-Tests erzielten Punktzahl wird zu max. 10% auf die Klausurpunktzahl angerechnet. Hiermit ist eine Verbesserung der Klausurbewertung um bis zu 0,7 Notenpunkte möglich. Die Anrechnung erfolgt nur, wenn Sie die Prüfung an sich mit der Mindestnote 4,0 bestanden haben. Der Bonus kann nur einmal im Prüfungszeitraum der Vorlesung angerechnet werden, entweder zum Haupttermin nach Vorlesungsende oder zum Nachholtermin im Folgesemester, wenn der Haupttermin nicht wahrgenommen wurde.
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • C.T. Chen. Control System Design, Pond Woods Press, 1987 • O. Föllinger. Regelungstechnik: Einführung in die Methoden und ihre Anwendung. 8. Auflage, Hüthig, 1994 • H. Geering. Regelungstechnik, 6. Auflage, Springer, 2004 • T. Kailath. Linear Systems, Prentice Hall, 1980 • G. Ludyk. Theoretische Regelungstechnik 1, Springer, 1995 • D.G. Luenberger. Introduction to Dynamic Systems, John Wiley & Sons, 1979 • J. Lunze. Regelungstechnik 1, 12. Auflage, Springer, 2020

- J. Lunze. Regelungstechnik 2, 10. Auflage, Springer, 2020
- L. Padulo, M.A. Arbib. System Theory, W.B. Saunders Company, 1974
- W.J. Rugh. Linear System Theory 2, Prentice Hall, 1996

NF Thermo- and Fluiddynamics

1	Modulbezeichnung 92476	Technische Thermodynamik I Technical thermodynamics	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Andreas Paul Fröba	
5	Inhalt	<p>Die Veranstaltung vertieft die Grundlagen der Technischen Thermodynamik und besitzt folgende inhaltliche Schwerpunkte:</p> <ul style="list-style-type: none"> • Grundbegriffe der Technischen Thermodynamik • Ideale Gase und deren Zustandsgleichungen • 1. und 2. Hauptsatz der Thermodynamik • Grenzen der Umwandlung von Energien • Thermodynamische Eigenschaften reiner Stoffe • Kreisprozesse • Ideale Gas- und Gas-Dampf-Gemische • Prozesse mit feuchter Luft 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • kennen die Begriffe und Grundlagen der Technischen Thermodynamik • erstellen energetische und exergetische Bilanzen • wenden thermodynamische Methodik für die Berechnung der Zustandseigenschaften sowie von Zustandsänderungen reiner Fluide an • berechnen relevante thermodynamische Prozesse und bewerten diese aufgrund charakteristischer Kennzahlen • optimieren thermodynamische Prozesse • lösen auch komplexe Fragestellungen der Technischen Thermodynamik 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 3	
9	Verwendbarkeit des Moduls	NF Thermo- and Fluidynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (120 Minuten)	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 165 h	
14	Dauer des Moduls	1 Semester	

15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none">• Vorlesungsskript• H.D. Baehr, S. Kabelac, Thermodynamik

1	Modulbezeichnung 94304	Technische Thermodynamik II Technical thermodynamics II	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Technische Thermodynamik (Vertiefung) für ET und CE (3 SWS) Übung: Technische Thermodynamik Vertiefung Übung (2 SWS)	3,5 ECTS 1,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Andreas Paul Fröba Dr.-Ing. Tobias Klein Paul Damp	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Andreas Paul Fröba	
5	Inhalt	<p>Das Modul Technische Thermodynamik - Vertiefung beinhaltet neben einer Wiederholung der Grundlagen zur Bilanzierung von Masse, Energie, Impuls, Entropie und Exergie die Themen Verbrennungstechnik, Strömungsprozesse und Einführung in die Gasdynamik, Kältetechnik sowie effiziente Wärmeübertragung. Das Thema Verbrennungsprozesse soll zugleich als allgemeine Einführung in die thermodynamische Behandlung von Systemen dienen, in denen chemische Reaktionen stattfinden. Schwerpunkte der energetischen Betrachtung von Verbrennungsprozessen bilden die Berechnung der freigesetzten Wärme sowie der Verbrennungstemperatur. Mit Hilfe von Entropiebilanzen wird die Effizienz von Verbrennungsprozessen in Form des exergetischen Wirkungsgrades bzw. in Form von auftretenden Exergieverlusten analysiert. Bei Strömungsprozessen sollen insbesondere kompressible Medien und somit auch Hochgeschwindigkeitsströmungen betrachtet werden, bei denen strömungsmechanische und thermodynamische Vorgänge stets miteinander verknüpft ablaufen. Hier werden neben den Grundgleichungen zur Modellierung von entsprechenden Strömungen und Zustandsänderungen spezielle Anwendungen von Düse und Diffusor diskutiert, z.B. im Bereich der Antriebstechnik und Kältetechnik. Das Thema Kältetechnik behandelt zunächst theoretisch deren Grundaufgaben. Schwerpunkte bilden dann unterschiedliche Verfahren und Anlagen zur Erzeugung von tiefen Temperaturen einschließlich derer zur Gasverflüssigung. Bei der Auslegung und Optimierung von Anlagen zur Erzeugung mäßig tiefer Temperaturen, z.B. in Form von Kompressions-, Dampfstrahl- und Absorptionskältemaschine, werden auch ökologische und ökonomische Kriterien bei Auswahl von Kältemitteln gegenübergestellt. Das Thema effiziente Wärmeübertragung beschäftigt sich insbesondere mit der wärme- und strömungstechnischen Auslegung von indirekten Wärmeübertragungssystemen. Für deren Optimierung werden Exgieverluste durch Druckänderung, Temperaturunterschiede, Mischung und Wärmeübertragung an die Umgebung betrachtet und diskutiert. Für den Fall der Kondensation wird das Verbesserungspotential eines gesteigerten Wärmeübertragungskoeffizienten zur Minimierung von Exergieverlusten mit Hilfe von Beispielen aus dem Bereich der Kraftwerkstechnik und Meerwasserentsalzung demonstriert.</p>	

6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> wenden wesentliche thermodynamische Grundlagen zur Konzeptionierung und Entwicklung von Systemen und Prozessen der Energie- und Verfahrenstechnik, darunter speziell solcher der Verbrennungs-, Strömungs-, Kälte- und Wärmetechnik an können Berechnungen zur thermodynamischen Optimierung analysieren und selbständig durchführen sowie die notwendigen Hilfsmittel methodisch angemessen anwenden diskutieren die Auslegung und Optimierung von Anlagen im Bereich der Wärme-, Energie- und Verfahrenstechnik unter Berücksichtigung ökologischer und ökonomischer Kriterien
7	Voraussetzungen für die Teilnahme	Grundkenntnisse der Technischen Thermodynamik
8	Einpassung in Studienverlaufsplan	Semester: 4;6
9	Verwendbarkeit des Moduls	NF Thermo- and Fluidynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	mündlich
11	Berechnung der Modulnote	mündlich (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 75 h Eigenstudium: 75 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>H. D. Baehr und S. Kabelac, Thermodynamik, Springer 2009 (14. Auflage)</p> <p>E. Hahne, Technische Thermodynamik, Oldenbourg 2004 (4. Auflage)</p> <p>K. Lucas, Thermodynamik, Springer 2000 (2. Auflage)</p> <p>D. Rist, Dynamik realer Gase, Springer 1996</p> <p>R. Günther, Verbrennung und Feuerungen, Springer 1984</p> <p>A. Bejan, Advanced Engineering Thermodynamics, John Wiley & Sons 1988</p>

1	Modulbezeichnung 97030	Wärme- und Stoffübertragung Heat and mass transfer	5 ECTS
2	Lehrveranstaltungen	Übung: Übung zu Wärme- und Stoffübertragung für ET (1 SWS) Vorlesung: Wärme- und Stoffübertragung für ET, MB und CE (2 SWS)	- -
3	Lehrende	Dr.-Ing. Franz Huber	

4	Modulverantwortliche/r	Dr.-Ing. Franz Huber Prof. Dr.-Ing. Stefan Will	
5	Inhalt	<ul style="list-style-type: none"> • Grundlagen der Wärme-, Stoff und Impulsübertragung • Wärmeleitung in ruhenden Körpern • Wärmeübertragung in einphasigen Strömungen durch konvektiven Wärmeübergang • Diffusion und Stoffübertragung an strömende Fluide • Analogie zwischen Wärme- und Stoffübertragung • Wärmeübertragung durch Strahlung • Wärmeübertragung bei Kondensation und Verdampfung • Wärmeübertrager 	
6	Lernziele und Kompetenzen	Die Studierenden: <ul style="list-style-type: none"> • verstehen die Mechanismen der Wärme- und Stoffübertragung und können ihre Bedeutung und ihren Einzelbeitrag bei technischen Problemstellungen ermitteln • können die Beiträge der verschiedenen Wärmeübertragungsmechanismen (Wärmeleitung, Konvektion, Strahlung und bei Phasenwechsel) quantifizieren • können die thermische Auslegung von einfachen Wärmeübertragern selbständig durchführen • verstehen die Analogie zwischen Wärme- und Stoffübertragung und sind in der Lage, sie bei der Lösung von Stoffübertragungsproblemen zu nutzen 	
7	Voraussetzungen für die Teilnahme	Empfohlen: Grundlegende Kenntnisse der Mathematik (Differential- und Integralrechnung, mathematische Charakterisierung von Feldern, Differentialoperatoren, gewöhnliche und partielle Differentialgleichungen) / Grundlagen der Thermodynamik	
8	Einpassung in Studienverlaufsplan	Semester: 6	
9	Verwendbarkeit des Moduls	NF Thermo- and Fluidynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (120 Minuten)	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Sommersemester	

13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 45 h Eigenstudium: 105 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none">• Vorlesungsskript• H. D. Baehr, K. Stephan, Wärme- und Stoffübertragung, Springer (2010)

NF Information Technology

1	Modulbezeichnung 43830	Einführung in die Grundlagen der Elektrotechnik für CE-Studierende Introduction to the Foundations of Electrical Engineering for CE Students	2,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Dr.-Ing. Clemens Stierstorfer	
5	Inhalt	<p>Inhaltsübersicht (jeweils nur grundlegende Einführungen):</p> <ul style="list-style-type: none"> • Elektrisches Feld, Elektrostatik, Kondensator • Strom und Spannung, Widerstand, Ohm'sches Gesetz • Analyse von Gleichstromnetzwerken • Magnetfeld, Induktion, Induktivitäten • Passive lineare Netzwerke • Komplexe Wechselstromrechnung, Analyse von monofrequent angeregten passiven Netzwerken • Übertragungsfunktion • Signale und deren Charakterisierung • Grundlagen von Halbleiterbauelementen • Transistor-Grundsaltungen • Operationsverstärker-Grundsaltungen • Module informationstechnischer Systeme und deren elektronische Implementierung • Prinzipielle Eigenschaften drahtgebundener und drahtloser Signalausbreitung 	
6	Lernziele und Kompetenzen	<ul style="list-style-type: none"> • Die Studierenden berechnen elektrische Felder für einfachste Ladungskonfigurationen und ermitteln hieraus Kraftwirkungen zwischen Ladungen • Sie verstehen die Begriffe Potential und Spannung • Sie verstehen, den Ladungsinhalt von , bzw. die Spannung an Kondensatoren zu berechnen • Sie haben Vorstellungen vom Fluss elektrischer Ladungen durch Leiter mit Widerstand • Die Studierenden nutzen die Ohmsche Gesetze zur Analyse von einfachen elektrischen Netzwerken • Sie begreifen magnetische Kraftwirkungen und können diese für einfache Konfigurationen berechnen • Sie kennen den Zusammenhang zwischen Stromfluss und magnetischen Feld • Sie begreifen das Prinzip der magnetischen Induktion • Sie kennen die grundlegenden Gleichungen zum Zusammenhang von Strom und Spannung beim Widerstand, der Spule und dem Kondensator, also bei den grundlegenden passiven Bauelementen der Elektrotechnik 	

		<ul style="list-style-type: none"> • Sie verstehen die Grundlagen der komplexen Wechselstromrechnung und analysieren einfache passive Netzwerke bei monofrequenter Anregung • Sie berechnen und nutzen Übertragungsfunktionen für einfache passive lineare Netzwerke • Sie verstehen Zwecke und Wirkungsweisen von Tief-, Hoch-, Band- und Allpässen als lineare Netzwerke • Sie verstehen die wichtigsten Parameter elektrischer Signale wie Leistung, Effektivwert, Spitzenwert, Crestfaktor, Bandbreite • Sie kennen Fourier-Reihe, Laplace- und Fourier-Transformation und verstehen das Konzept des Frequenzbereich, sowie Spektren und Bandbreite von Signalen • Sie verstehen die einfachsten physikalischen Grundlagen elektronischer Bauelemente • Sie verstehen die prinzipiellen Funktionsweisen von Dioden und Transistoren und den zugehörigen Grundschaltungen • Sie sind in der Lage, einfache Operationsverstärkergrundschaltungen zu dimensionieren • Sie verstehen die Funktionalität der Grundmodule Verstärker, Frequenzumsetzer, Filter, Wandler, etc. von nachrichtentechnischen Systemen • Sie verstehen die Prinzipien der Ausbreitung elektromagnetischer Felder, leitergebunden und drahtlos
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	NF Information Technology Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Studienleistung Jeder Vorlesungstermin wird mit einem kurzen Quiz auf StudOn kombiniert. Diese Quizze müssen erfolgreich gelöst werden (8 Quizze, davon müssen 7 bestanden werden).
11	Berechnung der Modulnote	Studienleistung (bestanden/nicht bestanden)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 15 h Eigenstudium: 60 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 93500	Digitale Signalverarbeitung Digital signal processing	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Dr.-Ing. Heinrich Löllmann	
5	Inhalt	<ul style="list-style-type: none"> • A/D and D/A conversion • <ul style="list-style-type: none"> ◦ Time-domain and z-domain representations ◦ Signal flow graphs ◦ Analytic computation of the frequency response ◦ Special systems (allpass, minimum phase, and linear phase systems) • Design of recursive and non-recursive filters • Multirate systems and filter banks • Frequency-domain signal analysis • Effects of finite wordlength 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • analysieren zeitdiskrete lineare zeitinvariante Systeme durch Ermittlung der beschreibenden Funktionen und Parameter • wenden grundlegende Verfahren zum Entwurf zeitdiskreter Systeme an und evaluieren deren Leistungsfähigkeit • verstehen die Unterschiede verschiedener Methoden zur Spektralanalyse und können damit vorgegebene Signale analysieren • verstehen die Beschreibungsmethoden von Multiraten-Systemen und wenden diese zur Beschreibung von Filterbänken an • kennen elementare Methoden zur Analyse von Effekten endlicher Wortlängen und wenden diese auf zeitdiskrete lineare zeitinvariante Systeme an. 	
7	Voraussetzungen für die Teilnahme	Der Kurs setzt Kenntnisse der grundlegenden Theorie der zeitdiskreten deterministischen Signale voraus wie sie in Vorlesungen wie Signale und Systeme II vermittelt werden.	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	NF Information Technology Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	<p>Klausur (90 Minuten) Schriftliche Prüfung von 90 Minuten Dauer. Für diese Prüfung sind folgende Hilfsmittel erlaubt: eine handschriftliche Formelsammlung im Umfang eines zweiseitigen DIN-A4-Blattes und ein nicht programmierbarer Taschenrechner.</p>	

		Die Antworten können entweder auf Englisch oder auf Deutsch gegeben werden.
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> • A.V. Oppenheim and R. W. Schafer: Discrete-Time Signal Processing, Prentice Hall • J.G. Proakis and D.G. Manolakis: Digital Signal Processing, Prentice Hall

1	Modulbezeichnung 93601	Information Theory and Coding / Informationstheorie und Codierung Information theory and coding	5 ECTS
2	Lehrveranstaltungen	Übung: Informationstheorie und Codierung - Übung (1 SWS) Vorlesung: Informationstheorie und Codierung (3 SWS)	- 5 ECTS
3	Lehrende	Prof. Dr.-Ing. Ralf Müller Johanna Fröhlich	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Ralf Müller	
5	Inhalt	<p>1. Introduction: binomial distribution, (7,4)-Hamming code, parity-check matrix, generator matrix</p> <p>2. Probability, entropy, and inference: entropy, conditional probability, Bayes law, likelihood, Jensens inequality</p> <p>3. Inference: inverse probability, statistical inference</p> <p>4. The source coding theorem: information content, typical sequences, Chebychev inequality, law of large numbers</p> <p>5. Symbol codes: unique decidability, expected codeword length, prefix-free codes, Kraft inequality, Huffman coding</p> <p>6. Stream codes: arithmetic coding, Lempel-Ziv coding, Burrows-Wheeler transform</p> <p>7. Dependent random variables: mutual information, data processing lemma</p> <p>8. Communication over a noisy channel: discrete memory-less channel, channel coding theorem, channel capacity</p> <p>9. The noisy-channel coding theorem: jointly-typical sequences, proof of the channel coding theorem, proof of converse, symmetric channels</p> <p>10. Error-correcting codes and real channels: AWGN channel, multivariate Gaussian pdf, capacity of AWGN channel</p> <p>11. Binary codes: minimum distance, perfect codes, why perfect codes are bad, why distance isnt everything</p> <p>12. Message passing: distributed counting, path counting, low-cost path, min-sum (=Viterbi) algorithm</p> <p>13. Exact marginalization in graphs: factor graphs, sum-product algorithm</p> <p>14. Low-density parity-check codes: density evolution, check node degree, regular vs. irregular codes, girth</p> <p>15. Lossy source coding: transform coding and JPEG compression</p> <p>--</p> <p>1. Einleitung: Binomialverteilung, (7,4)-Hamming-Code, Paritätsmatrix, Generatormatrix</p> <p>2. Wahrscheinlichkeit, Entropie und Inferenz: Entropie, bedingte Wahrscheinlichkeit, Bayessches Gesetz, Likelihood, Jensensche Ungleichung</p> <p>3. Inferenz: Inverse Wahrscheinlichkeit, statistische Inferenz</p> <p>4. Das Quellencodierungstheorem: Informationsgehalt, typische Folgen, Tschebyschevsche Ungleichung, Gesetz der großen Zahlen</p>	

		<p>5. Symbolcodes: eindeutige Dekodierbarkeit, mittlere Codewortlänge, präfixfreie Codes, Kraftsche Ungleichung, Huffmancodierung</p> <p>6. Stromcodes: arithmetische Codierung, Lempel-Ziv-Codierung, Burrows-Wheeler-Transformation</p> <p>7. Abhängige Zufallsvariablen: Transinformation, Datenverarbeitungslemma</p> <p>8. Kommunikation over gestörte Kanäle: diskreter gedächtnisloser Kanal, Kanalcodierungstheorem, Kanalkapazität</p> <p>9. Das Kanalcodierungstheorem: verbundtypische Folgen, Beweis des Kanalcodierungstheorems, Beweis des Umkehrsatzes, symmetrische Kanäle</p> <p>10. Fehlerkorrigierende Codes und reale Kanäle: AWGN-Kanal, mehrdimensionale Gaußsche WDF, Kapazität des AWGN-Kanals</p> <p>11. Binäre Codes: Minimaldistanz, perfekte Codes, Warum perfekte Codes schlecht sind, Warum Distanz nicht alles ist</p> <p>12. Nachrichtenaustausch: verteiltes Zählen, Pfadzählen, günstigster Pfad, Minimumsummenalgorithmus</p> <p>13. Exakte Marginalisierung in Graphen: Faktorgraph, Summenproduktalgorithmus</p> <p>14. LDPC-Codes: Dichteevolution, Knotenordnung, reguläre und irreguläre Codes, Graphumfang</p> <p>15. Verlustbehaftete Quellencodierung: Transformationscodierung und JPEG-Kompression</p>
6	<p>Lernziele und Kompetenzen</p>	<p>The students apply Bayesian inference to problems in both communications and everyday's life.</p> <p>The students explain the concept of digital communications by means of source compression and forward-error correction coding.</p> <p>For the design of communication systems, they use the concepts of entropy and channel capacity.</p> <p>They calculate these quantities for memoryless sources and channels.</p> <p>The students proof both the source coding and the channel coding theorem.</p> <p>The students compare various methods of source coding with respect to compression rate and complexity.</p> <p>The students apply source compression methods to measure mutual information.</p> <p>The students factorize multivariate functions, represent them by graphs, and marginalize them with respect to various variables.</p> <p>The students explain the design of error-correcting codes and the role of minimum distance.</p> <p>They decode error-correcting codes by means of maximum-likelihood decoding and message passing.</p> <p>The students apply distributed algorithms to problems in both communications and everyday's life.</p> <p>The students improve the properties of low-density parity-check codes by widening the girth and/or irregularity in the degree distribution.</p> <p>The students transform source images into the frequency domain to improve lossy compression.</p> <p>--</p>

		<p>Die Studierenden wenden Bayessche Inferenz auf Probleme in der Nachrichtentechnik und im Alltagsleben an.</p> <p>Die Studierenden erklären die konzeptuelle Trennung von digitaler Übertragung in Quellen- und Kanalcodierung.</p> <p>Kommunikationssysteme entwerfen sie unter Betrachtung von Entropie und Kanalkapazität.</p> <p>Sie berechnen diese Größen für gedächtnislose Quellen und Kanäle.</p> <p>Die Studierenden beweisen sowohl das Quellen- als auch das Kanalcodierungstheorem.</p> <p>Die Studierenden vergleichen verschiedenartige Quellencodierungsverfahren hinsichtlich Komplexität und Kompressionsrate.</p> <p>Die Studierenden verwenden Quellencodierverfahren zur Messung von Transinformation.</p> <p>Die Studierenden faktorisieren Funktionen mehrerer Veränderlicher, stellen diese als Graph dar und marginalisieren sie bezüglich mehrerer Veränderlicher.</p> <p>Die Studierenden erklären den Entwurf von Kanalcodes und den Einfluss der Minimaldistanz.</p> <p>Sie decodieren Kanalcodes gemäß maximaler Likelihood und Nachrichtenaustausch.</p> <p>Die Studierenden wenden verteilte Algorithmen auf Probleme der Nachrichtentechnik und des Alltagslebens an.</p> <p>Die Studierenden verbessern die Eigenschaften von LDPC-Codes durch Erhöhung des Umfangs und/oder durch irreguläre Knotenordnungsverteilungen.</p> <p>Die Studierenden transformieren Bildquellen zur Verbesserung verlustbehafteter Kompression in den Frequenzbereich.</p>
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	NF Information Technology Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	in jedem Semester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	MacKay, D.: Information Theory, Inference, and Learning Algorithms, Cambridge University Press, Cambridge, 2003.

NF Mechatronics

1	Modulbezeichnung 92520	Elektromagnetische Felder I Electromagnetic fields I	2,5 ECTS
2	Lehrveranstaltungen	Vorlesung: Elektromagnetische Felder I (2 SWS) Tutorium: Tutorium zu Elektromagnetische Felder I (2 SWS)	2,5 ECTS -
3	Lehrende	Dr.-Ing. Gerald Gold Prof. Dr.-Ing. Klaus Helmreich	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Klaus Helmreich	
5	Inhalt	<p>Im ersten Teil der Vorlesung "Elektromagnetische Felder" wird zuerst der Begriff "Feld" eingeführt, die speziell damit verbundenen mathematischen Methoden und Aussagen sowie die zugrundeliegenden physikalischen Konzepte.</p> <p>Anschließend wird die Formulierung der Grundaussagen der elektromagnetischen Feldtheorie aus Experimenten und theoretischen Überlegungen in heutiger mathematischer Darstellung nachvollzogen. Dabei werden historische und aktuelle Begriffsbildungen einander gegenübergestellt - Atombau der Materie und Relativität waren bei Aufstellung der Theorie noch nicht bekannt!</p> <p>Das Nachvollziehen des historischen Begriffsbildungs- und Erkenntnisprozesses erleichtert den Zugang zur Begrifflichkeit und mathematischen Formulierung der Theorie und damit deren Verständnis und Vorstellbarkeit".</p> <p>In Kenntnis von Atombau der Materie und Relativität präzisiert die aktuelle Darstellung die Begriffe, wodurch deren Zahl reduziert werden kann.</p> <p>Folgerungen aus der Theorie werden vorgestellt - insbesondere die Existenz elektromagnetischer Wellen und die Deutung von Licht als solcher. Exemplarisch werden wesentliche Eigenschaften eines technisch besonders relevanten Wellentyps - der ebenen harmonischen Welle - abgeleitet.</p> <p>Phänomene in Materie im elektromagnetischen Feld werden aus atomistischer Sicht behandelt, was - zusammen mit der Festlegung der Maßeinheiten - zur aktuellen Begriffsbildung und Formulierung der Maxwell'schen Gleichungen (MG) führt.</p> <p>Daraus wird das Verhalten von Feldern an Materialübergängen abgeleitet.</p> <p>Als allgemeine Lösung der MG werden die elektromagnetischen Potentiale hergeleitet, ihre grundlegenden Eigenschaften erläutert und ihre Anwendung zur Lösung feldtheoretischer Fragestellungen dargestellt.</p> <p>Inhalt und Gültigkeitsbereich der Theorie werden diskutiert.</p> <p>Die Behandlung zeitlich konstanter elektrischer, magnetischer und Strömungsfelder - ihrer Entstehung und ihrer Eigenschaften - bildet den Abschluß des ersten Teils der Vorlesung.</p>	

		<p>In den Übungen wird der Stoff der Vorlesung durch die Anwendung auf konkrete wissenschaftliche und technische Problemstellungen und beispielartige Lösung von Standardproblemen vertieft. Weiteres Ziel der Übungen ist die Vorbereitung auf die schriftliche Prüfung.</p> <p>Inhaltsübersicht:</p> <ul style="list-style-type: none"> • Felder: Physikalische Konzepte und mathematische Beschreibung • Begriffe und Grundaussagen der elektromagnetischen Feldtheorie • Folgerungen aus den Grundaussagen: Ausblick auf elektromagnetische Wellen • Materie im Feld und Felder an Materialübergängen • Die Potentiale des elektromagnetischen Felds • Inhalt und Gültigkeitsbereich der elektromagnetischen Feldtheorie • Zeitunabhängige Felder, Teil 1
6	Lernziele und Kompetenzen	<p>Nach der Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage:</p> <ul style="list-style-type: none"> • Begriffe und physikalische Konzepte der elektromagnetischen Feldtheorie zu erklären • Vektoralgebraische und vektoranalytische Beziehungen und Umformungen zu verstehen und letztere auch vorzunehmen • Kraftwirkungen im elektromagnetischen Feld zu verstehen und zu berechnen • die Bedeutung von Feldgleichungen und Kontinuitätsgleichung zu verstehen • Induktionsvorgänge zu verstehen und für einfache Situationen zu berechnen • grundlegende Eigenschaften ebener elektromagnetischer Wellen zu beschreiben • Phänomene elektrischer und magnetischer Felder in Materie und an Materialübergängen zu verstehen und zu beschreiben • Felder und Potentiale einfacher Ladungs- und Stromdichteverteilungen z.B. mittels der Maxwell'schen Gleichungen, allgemeiner Lösungen der Poissongleichung oder aufgrund mathematischer Korrespondenzen zu berechnen • den Gültigkeitsbereich der Theorie zu benennen
7	Voraussetzungen für die Teilnahme	Voraussetzung: Vektoranalysis, z.B. aus der Mathematik-VL im Grundstudium
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	NF Mechatronics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (60 Minuten)

11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 30 h Eigenstudium: 45 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • Skript zur Vorlesung • Übungsaufgaben mit Lösungen auf der Homepage • Formelsammlung

1	Modulbezeichnung 92560	Grundlagen der Elektrotechnik I Foundations of electrical engineering I	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Bernd Witzigmann	
5	Inhalt	<p>Diese Vorlesung bietet einen Einstieg in die physikalischen Grundlagen der Elektrotechnik und Elektronik. Ausgehend von beobachtbaren Kraftwirkungen zwischen Ladungen und zwischen Strömen wird der Begriff des elektrischen und magnetischen Feldes eingeführt. Mit den daraus abgeleiteten integralen Größen Spannung, Strom, Widerstand, Kapazität und Induktivität wird das Verhalten der passiven Bauelemente diskutiert. Am Beispiel der Gleichstromschaltungen werden die Methoden der Netzwerkanalyse eingeführt und Fragen nach Wirkungsgrad und Zusammenschaltung von Quellen untersucht. Einen Schwerpunkt bildet das Faradaysche Induktionsgesetz und seine Anwendungen. Die Bewegungsinduktion wird im Zusammenhang mit den Drehstromgeneratoren betrachtet, die Ruheinduktion wird sehr ausführlich am Beispiel der Übertrager und Transformatoren diskutiert. Einen weiteren Schwerpunkt bildet die Behandlung zeitlich periodischer Vorgänge. Die komplexe Wechselstromrechnung bei sinusförmigen Strom- und Spannungsformen wird ausführlich behandelt.</p> <ol style="list-style-type: none"> 1. Physikalische Grundbegriffe 2. Das elektrostatische Feld 3. Das stationäre elektrische Strömungsfeld 4. Einfache elektrische Netzwerke 5. Das stationäre Magnetfeld 6. Das zeitlich veränderliche elektromagnetische Feld 7. Wechselspannung und Wechselstrom 	
6	Lernziele und Kompetenzen	<p>Nach der Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage:</p> <ul style="list-style-type: none"> • den Begriff des Feldes zu verstehen, • Gleich- und Wechselstromschaltungen mit Widerständen, Kapazitäten, Induktivitäten und Transformatoren zu entwickeln, • Schwingkreise und Resonanzerscheinungen zu analysieren, • Energie- und Leistungsberechnungen durchzuführen, • Schaltungen zur Leistungsanpassung und zur Blindstromkompensation zu bewerten, • das Drehstromsystem zu verstehen. 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 3	

9	Verwendbarkeit des Moduls	NF Mechatronics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (120 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • M. Albach, Elektrotechnik, Pearson Verlag • Manfred Albach: Grundlagen der Elektrotechnik I und II, Pearson-Verlag • Übungsaufgaben mit Lösungen auf der Homepage • Optional: Übungsbuch, Pearson-Verlag

1	Modulbezeichnung 92580	Grundlagen der Elektrotechnik III Foundations of electrical engineering III	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Philipp Beckerle	
5	Inhalt	<ul style="list-style-type: none"> • Umfang und Bedeutung der elektrischen Messtechnik • Die Grundlagen des Messens • Fourier-Transformation • Laplace-Transformation • Netzwerkanalyse im Zeit- und Laplace-Bereich • Übertragungsfunktion und Bode-Diagramm • Nichtlineare Bauelemente, Schaltungen und Systeme • Operationsverstärker • Messverstärker • Messfehler • Messung von Gleichstrom und Gleichspannung • Ausschlagbrücken • Abgleichbrücken, Messung von elektrischen Impedanzen 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • ordnen die behandelten Verfahren gemäß ihrer Eignung für spezifische Probleme (Zeit-/Frequenzbereich, Linear/ Nichtlinear) ein. • wählen geeignete Verfahren zur Analyse elektrischer Netzwerke aus und wenden diese an. • interpretieren die Ergebnisse und zeigen Zusammenhänge zwischen den Lösungsverfahren auf. • kennen einfache Grundsaltungen mit Operationsverstärkern und sind in der Lage, diese zu analysieren. • kennen die behandelten Messschaltungen und ihre Einsatzmöglichkeiten. • analysieren Brückenschaltungen. • wenden grundlegende Konzepte der Messfehlerrechnung auf Messschaltungen an. • reflektieren selbstständig den eigenen Lernprozess und nutzen die Präsenzzeit zur Klärung der erkannten Defizite. 	
7	Voraussetzungen für die Teilnahme	Grundlagen der Elektrotechnik I und II	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	NF Mechatronics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)	

11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Lehrbuch: Elektrische Messtechnik", R. Lerch, 7. Aufl. 2016, Springer-Verlag Übungsbuch: Elektrische Messtechnik Übungen", R. Lerch, M. Kaltenbacher, F. Lindinger, A. Sutor, 2. Aufl. 2005, Springer-Verlag

NF Computational Optics

1	Modulbezeichnung 48311	Modern Optics 1: Advanced Optics Modern optics 1: Advanced optics	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Stephan Götzinger	
5	Inhalt	<p>Scalar wave optics: Maxwell equations, solutions to the wave equation, interference effects;</p> <p>Fourier optics: propagation in free space, through aperture and lenses, Fourier transformation in the far field;</p> <p>Vectorial wave optics: Maxwell equation and solution of the vectorial fields: Gaussian laser beam (fundamental and higher order modes), focusing of vector fields in free space, vector fields with optical angular momentum;</p> <p>Optics in waveguides: geometrical approach and Maxwell equation with boundary conditions, transverse modes, cutoff for planar waveguide, optical fibers, tapers, couplers;</p> <p>Whispering gallery mode resonators: modal description, applications.</p>	
6	Lernziele und Kompetenzen	The students will get exposed to more advanced optical topics ranging from thin periodic layers, optical cavities and waveguides to optical fibers, plasmonics, metamaterials, angular momentum of light and modern microscopy techniques. They will also apply newly introduced methods to specific examples.	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	NF Computational Optics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	mündlich (30 Minuten) PL: Oral examination 30 min.	
11	Berechnung der Modulnote	mündlich (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Englisch	
16	Literaturhinweise	<p>Christopher Foot: Atomic physics</p> <p>Saleh Teich: Fundamentals of Photonics</p>	

Mark Fox: Quantum Optics: an introduction

1	Modulbezeichnung 66120	Experimentalphysik 3: Optik und Quanteneffekte Experimental physics 3: Optics and quantum effects	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Peter Hommelhoff Prof. Dr. Christoph Marquardt Prof. Dr. Vahid Sandoghdar
5	Inhalt	<p>Optik und Quantenphänomene</p> <ul style="list-style-type: none"> • Licht als Welle: Belege für die Wellennatur des Lichts, Herleitung der Wellengleichung aus den Maxwell-Gleichungen, Lösungen in Form von ebenen Wellen, Kugelwellen, monochromatische Felder • Licht und Materie: Einzelstreuer (getriebener Dipol, Lichtstreuung), Feldausbreitung im homogenen Material, Polarisierung und Stromdichte, modifizierte Maxwell-Gleichungen, modifizierte Wellengleichung, Stetigkeitsbedingungen an Grenzflächen, Brechungsgesetz, Fresnelformeln, Brewsterwinkel, Totalreflexion, frustrierte Totalreflexion und Tunneleffekt bei Licht, Polarisierung des Materials (Suszeptibilität, Dispersion) • Geometrische Optik: Strahlenoptik, Matrizenoptik (Prinzip, Anwendung auf Linsen, Abbildungen), Hauptebenen, Abbildungsfehler (chromatische Aberrationen, Fehler für monochromatische Wellen), optische Resonatoren • Beugung und Interferenz: Ausbreitungsgleichung unter Randbedingungen, Huygenssches Prinzip, Fraunhoferbeugung (Entstehung des Beugungsbildes, Beugungsbilder, Grenzen), Mikroskope, Teleskope, Auflösungsgrenzen, Abbildungstechniken, das Auge. Polarisierung elektromagnetischer Felder, Ebene Wellen im homogenen Material, Polarisierungsformen von Licht, Polarisationsphänomene im durchstrahlten Material, Doppelbrechung, polarisierende Elemente • Grundlegende Experimente zu Quantenphänomenen: Teilchencharakter des Lichts, äußerer lichtelektrischer Effekt (Photoeffekt), Hohlraumstrahlung nach Planck, Compton-Effekt, Wellencharakter von Teilchen (Elektronenbeugung, Streuung im Kristall), Konsequenzen der Wellennatur des Elektrons • Grundgleichungen der Quantenmechanik: Schrödinger-Gleichung, zeitunabhängige Schrödinger-Gleichung, Interpretation der quantenmechanischen Wellenfunktion, Kastenpotenzial, Tunneleffekt mit Materiewellen

6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • erläutern und erklären die experimentellen Grundlagen und die quantitativ-mathematische Beschreibung der Optik und von Quantenphänomenen gemäß den detaillierten Themen im Inhaltsverzeichnis • wenden die physikalischen Gesetze und jeweiligen mathematischen Methoden auf konkrete Problemstellungen an
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	NF Computational Optics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	mündlich (30 Minuten)
11	Berechnung der Modulnote	mündlich (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 92390	Photonik 1 Photonics 1	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Bernhard Schmauß	
5	Inhalt	Es werden umfassend die technischen und physikalischen Grundlagen des Lasers behandelt. Der Laser als optische Strahlquelle stellt eines der wichtigsten Systeme im Bereich der optischen Technologien dar. Ausgehend vom Helium-Neon-Laser als Beispielsystem werden die einzelnen Elemente wie aktives Medium und Resonatoren eines Lasers sowie die ablaufenden physikalischen Vorgänge eingehend behandelt. Es folgt die Beschreibung von Laserstrahlen und ihrer Ausbreitung als Gauß-Strahlen sowie Methoden zur Beurteilung der Strahlqualität. Eine Übersicht über verschiedene Lasertypen wie Gaslaser, Festkörperlaser und Halbleiterlaser bietet einen Einblick in deren charakteristische Eigenschaften und Anwendungen. Vervollständigt wird die Vorlesung durch die grundlegende Beschreibung von Lichtwellenleitern, Faserverstärkern und halbleiterbasierten optoelektronischen Bauelementen wie Leuchtdioden und Photodioden.	
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • können Grundlagen der Physik des Lasers darlegen. • verstehen Eigenschaften und Beschreibungsmethoden von laseraktiven Medien, der stimulierte Strahlungsübergänge, der Rategleichungen, von optischen Resonatoren und von Gauß-Strahlen. • können verschiedene Lasertypen aus dem Bereichen Gaslaser, Festkörperlaser und Halbleiterlaser erklären und vergleichen. • können grundlegende Eigenschaften von Lichtwellenleiter und Lichtwellenleiterbauelementen erklären und skizzieren. • verstehen Aufbau und Funktionsweise ausgewählter optoelektronischer Bauelemente. • können grundlegende Fragestellung der Lasertechnik eigenständig bearbeiten, um Laserstrahlquellen weiterzuentwickeln und Lasertechnik und Photonik in einer Vielzahl von Anwendungen in Bereichen wie Medizintechnik, Messtechnik, Übertragungstechnik, Materialbearbeitung oder Umwelttechnik einzusetzen. 	
7	Voraussetzungen für die Teilnahme	Empfohlen werden Kenntnisse im Bereich: <ul style="list-style-type: none"> • Experimentalphysik, Optik • Elektromagnetische Felder • Grundlagen der Elektrotechnik 	
8	Einpassung in Studienverlaufsplan	Semester: 3	

9	Verwendbarkeit des Moduls	NF Computational Optics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>Eichler, J., Eichler, H.J: Laser. 7. Auflage, Springer Verlag, Berlin 2010.</p> <p>Reider, G.A.: Photonik. 3. Auflage, Springer Verlag, Berlin 2012.</p> <p>Bergmann, Schäfer: Lehrbuch der Experimentalphysik, Bd.3: Optik. DeGruyter 2004.</p> <p>Saleh, B., Teich, M.C.: Grundlagen der Photonik. 2. Auflage, Wiley-VCH 2008.</p> <p>Träger, F. (Editor): Springer Handbook of Lasers and Optics, 2. Auflage, Springer Verlag, Berlin 2012.</p>

1	Modulbezeichnung 96350	Photonik 2 Photonics 2	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Photonik 2 (2 SWS) Übung: Photonik 2 Übung (2 SWS)	5 ECTS -
3	Lehrende	Prof. Dr.-Ing. Bernhard Schmauß Jasper Freitag	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Bernhard Schmauß	
5	Inhalt	<p>Aufbauend auf "Photonik 1" werden fortgeschrittene Verfahren der Laser-Messtechnik, komplexe Laser-Systeme sowie deren technische Anwendungen besprochen.</p> <p>In einem ersten Themenkomplex werden Messverfahren für praktisch wichtige Laserkenngrößen wie z.B. Laserstrahlleistung, Polarisationszustand und Spektrum der Lichtwelle behandelt. Anschließend wird die räumliche und zeitliche Kohärenz eines Laserstrahls diskutiert. Dies ist die Grundlage für interferometrische Messverfahren zur Bestimmung von Lichtwellenlängen und hochaufgelösten optischen Spektren oder auch für mechanische Größen wie Weg und Winkelbeschleunigung.</p> <p>Rauschquellen in photonischen Systemen werden beschrieben und diskutiert. Wichtige Maßnahmen zur Reduktion von Rauschen in optischen Aufbauten werden vorgestellt.</p> <p>Optische Verstärker auf Glasfaserbasis, sog. Faserverstärker und darauf aufbauende Faserlaser werden in einem eigenen Kapitel vorgestellt. Faser-Bragg-Gitter als wichtige Bestandteile eines Faserlasers werden in Herstellung und Anwendung. U.a. in der Messtechnik diskutiert.</p> <p>Zeitlich dynamische Vorgänge im Laser, beschrieben durch die so genannten Rategleichungen und deren Lösung, werden ausführlich behandelt. Begriffe wie Spiking oder Relaxationsschwingungen und Verfahren wie Mode-Locking oder Q-Switching werden besprochen. Daraus wird die Funktion und die technische Anwendung von Lasern zur Erzeugung von energiereichen Lichtimpulsen bis hin zu sogenannten Femtosekundenlasern abgeleitet.</p> <p>Das Themengebiet der optischen Frequenzumsetzung wird mit einem Kapitel zur linearen und nichtlinearen Optik eingeleitet. Technische Anwendungen wie optische Frequenzverdoppelung, Erzeugung von UV-Licht durch Frequenzvervielfachung werden darauf aufbauend besprochen. Ein Kapitel zum Raman-Effekt und zur stimulierten Brillouin-Streuung sowie deren Anwendung schließt den Inhalt ab.</p> <p>Methoden und Systeme aus "Photonik 2" werden eingesetzt z.B. für die Präzisionsmesstechnik, in der industriellen Materialbearbeitung, in der Bioanalytik, für die Medizintechnik, in Geräten der Unterhaltungselektronik oder in der optischen Nachrichtentechnik.</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • besitzen spezialisiertes und vertieftes Wissen über Laser und den in den Inhalten beschriebenen photonischen Systemen und Methoden. 	

		<ul style="list-style-type: none"> • können die im Inhalt beschriebenen fortgeschrittenen Methoden der Photonik erklären und anwenden. • können technische und wissenschaftliche Anwendungen dieser photonischen Systeme diskutieren, beurteilen und vergleichen. • sind in der Lage, derartige photonische Systeme zu konzipieren und zu entwickeln. • können eigenständige Ideen und Konzepte zur Lösung wissenschaftlicher und beruflicher Probleme der Photonik entwickeln.
7	Voraussetzungen für die Teilnahme	<ul style="list-style-type: none"> • Photonik 1 oder vergleichbare Grundlagen der Photonik und Lasertechnik.
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	NF Computational Optics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	mündlich (30 Minuten)
11	Berechnung der Modulnote	mündlich (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>Eichler, J., Eichler, H.J: Laser. Springer Verlag, Berlin 2006.</p> <p>Reider, G.A.: Photonik. 2. Auflage, Springer Verlag, Berlin 2005.</p> <p>Bergmann, Schäfer: Lehrbuch der Experimentalphysik, Bd.3: Optik. DeGruyter 1993.</p> <p>Demtröder, W: Laserspektroskopie. Springer Verlag, Berlin 2000.</p>

NF Solid Mechanics and Dynamics

1	Modulbezeichnung 43840	Computational Engineering 2 Computational engineering 2	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Computational Engineering 2 (2 SWS) Übung: Computational Engineering 2 - Übung (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Dietmar Fey Prisca Rambach	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Dietmar Fey	
5	Inhalt	<p>Die Simulation hat sich in den letzten Jahrzehnten immer stärker in den Naturwissenschaften und im Ingenieurwesen etabliert. Besonders attraktiv sind Simulationen in Bereichen, wo Experimente teuer, schwierig, gefährlich oder gar unmöglich sind. Der Fokus dieser Veranstaltung liegt besonders auf der Berechnung und Implementierung einfacher physikalischer Modelle.</p> <p>Das Modul soll Hintergrundwissen und Modelle zur Simulation am Computer durchgeführter Experimente und Problemstellungen aus den Ingenieur- und Naturwissenschaften vermitteln. Die Übungen sind dazu da, die Methoden am Computer zu implementieren und damit zu "experimentieren".</p> <p>Ein weiteres Ziel ist, den Studierenden des Studiengangs Computational Engineering einen Einblick in die technischen Anwendungsfächer (TAFs) zu geben. Dazu werden Vertreter aus den verschiedenen Anwendungsfächern eingeladen.</p> <p>Folgende Inhalte werden adressiert:</p> <ul style="list-style-type: none"> • Gleichungslöser • ADAS-Algorithmen • Einfache Beispiele aus der Astrophysik • Performance-Optimierung von Programmen • Umsetzung auf Parallelrechnern 	
6	Lernziele und Kompetenzen	<p>Die Studierenden sollen</p> <ul style="list-style-type: none"> • einen Einblick in die Inhalte der technischen Anwendungsfächer (TAFs) bekommen • in die Lage versetzt werden, einfache physikalische Modelle zu verstehen und zu berechnen • die eingeführten Methoden am Computer selbständig implementieren und damit "experimentieren" 	
7	Voraussetzungen für die Teilnahme	Hintergrundwissen in einer höheren Programmiersprache (bevorzugt Java)	
8	Einpassung in Studienverlaufsplan	Semester: 2	
9	Verwendbarkeit des Moduls	<p>NF Automatic Control Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010</p> <p>NF Computational Optics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010</p> <p>NF Information Technology Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010</p>	

		NF Mechatronics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Solid Mechanics and Dynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Thermo- and Fluidynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Übungsleistung schriftlich/mündlich (30 Minuten)
11	Berechnung der Modulnote	Übungsleistung (bestanden/nicht bestanden) schriftlich/mündlich (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 45 h Eigenstudium: 105 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Bungartz, Zimmer, Buchholz, Pflüger: Modellbildung und Simulation. Springer, ISBN 978-3-540-79809-5

1	Modulbezeichnung 66621	Experimentalphysik für Naturwissenschaftler I Experimental physics for natural scientists I	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Stephan Götzinger apl. Prof. Dr. Norbert Lindlein	
5	Inhalt	<ul style="list-style-type: none"> • Einführung in die Experimentalphysik: Erkenntnisprozesse und Methoden der modernen Physik, Struktur der Materie, Wechselwirkungen, Einteilung der Physik in Teilgebiete, physikalische Größen: SI System, Messgenauigkeit, Messfehler • Mechanik: Punktmechanik, Mechanik starrer Körper, Schwingungen und Wellen, Mechanik von Flüssigkeiten und Gasen, Strömungsmechanik • Wärmelehre: Grundlagen, Hauptsätze der Wärmelehre, Wärmetransport, Phasenübergänge • Vertiefung und Ergänzung der Vorlesungsinhalte durch Übungsaufgaben 	
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • erklären die Grundlagen der Experimentalphysik aus dem Bereich der Mechanik und grundlegender Wärmelehre • wenden statistische Methoden zur Fehlerabschätzung der Messergebnisse an • setzen die Vorlesungsinhalte mit Hilfe thematisch passender Übungsaufgaben praktisch um. 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 1	
9	Verwendbarkeit des Moduls	NF Automatic Control Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Computational Optics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Information Technology Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Mechatronics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Solid Mechanics and Dynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Thermo- and Fluidynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)	

11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 75 h Eigenstudium: 75 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	D. Halliday, R. Resnick, J. Walker, "Physik", Wiley-VCH P.A. Tipler, "Physik", Spektrum Akad. Verlag J. Orear, "Physik", Hanser Fachbuch Verlag E. Hering, R. Martin, M. Stohrer, "Physik für Ingenieure", Springer W. Demtröder, "Experimentalphysik 1-Mechanik und Wärme", Springer

1	Modulbezeichnung 66631	Experimentalphysik für Naturwissenschaftler II Experimental physics for natural scientists II	5 ECTS
2	Lehrveranstaltungen	Übung: Übungen zur Experimentalphysik für Naturwissenschaftler II (1 SWS) Vorlesung: Experimentalphysik für Naturwissenschaftler II (4 SWS)	- 5 ECTS
3	Lehrende	apl. Prof. Dr. Norbert Lindlein	

4	Modulverantwortliche/r	Prof. Dr. Stephan Götzinger apl. Prof. Dr. Norbert Lindlein
5	Inhalt	<p>*I. Elektrizitätslehre*</p> <ol style="list-style-type: none"> 1. Einführung: Feldbegriff, elektrische Ladung, Ladungstransport, Stromstärke, Spannung, Widerstand, Ohmsches Gesetz 2. Zeitunabhängige elektrische Felder, Quellen statischer elektrischer Felder, Plattenkondensator, Kapazität, Materie im elektrischen Feld 3. Zeitunabhängige magnetische Felder, Erzeugung magnetischer Felder, Lorentzkraft, magnetische Flußdichte, magnetischer Fluß, Materie im Magnetfeld: Dia-, Para-, Ferromagnetismus 4. Zeitabhängige elektromagnetische Felder, Magnetische Induktion, Lenzsche Regel, zeitlich veränderliches elektrisches Feld Elektronenröhre 5. Wechselstrom, Wechselstromwiderstände, elektrische Leistung, elektrische Schwingkreise, Effektivwerte für Strom und Spannung 6. Elektromagnetische Wellen, Wellengleichungen, Hertzscher Dipol, weitere Wellenerscheinungen <p>*II. Optik*</p> <ol style="list-style-type: none"> 1. Geometrische Optik: Natur des Lichts, Brechung und Reflexion des Lichts, Abbildung durch Linsen, optische Instrumente 2. Wellenoptik: Kohärenz, Interferenz, Beugung an Spalt und Gitter, Auflösungsvermögen von Fernrohr und Mikroskop, Interferometer, polarisiertes Licht, Doppelbrechung, Streuung und Absorption von Licht 3. Quantenoptik: Licht als Teilchen, Photoeffekt, Comptoneffekt, Röntgenstrahlung, Plancksches Strahlungsgesetz 4. Materiewellen: Elektronen als Welle, Elektronenbeugung, De Broglie Wellenlänge <p>*III. Atomphysik*</p> <ol style="list-style-type: none"> 1. Franck-Hertz Versuch, Bohr'sches Atommodell 2. Wasserstoffatom, Schalenmodell, elektromagnetische Übergänge <p>*IV. Kernphysik*</p> <ol style="list-style-type: none"> 1. Kernaufbau, Bindungsenergie, Tröpfchenmodell 2. Radioaktive Strahlung 3. Kernspaltung 4. Kernfusion <p>*V. Teilchenphysik</p>
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • stellen grundlegende Prinzipien zum Elektromagnetismus, zur Optik und zur Atomphysik dar

		<ul style="list-style-type: none"> • setzen die Vorlesungsinhalte mit Hilfe thematisch passender Übungsaufgaben praktisch um.
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 2
9	Verwendbarkeit des Moduls	NF Automatic Control Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Computational Optics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Information Technology Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Mechatronics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Solid Mechanics and Dynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 NF Thermo- and Fluidynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 75 h Eigenstudium: 75 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Paul A. Tipler and Gene Mosca, Physik für Wissenschaftler und Ingenieure (7. Auflage), Springer, ISBN 978-3-642-54166-7 (eBook)

1	Modulbezeichnung 94500	Dynamik starrer Körper Dynamics of rigid bodies	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Sigrid Leyendecker	
5	Inhalt	<ul style="list-style-type: none"> • Kinematik von Punkten und starren Körpern • Relativkinematik von Punkten und starren Körpern • Kinetik des Massenpunktes • Newton'sche Axiome • Energiesatz • Stoßvorgänge • Kinetik des Massenpunktsystems • Lagrange'sche Gleichungen 2. Art • Kinetik des starren Körpers • Trägheitstensor • Kreiselgleichungen • Schwingungen 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • sind vertraut mit den grundlegenden Begriffen und Axiomen der Dynamik; • können Bewegungen von Massepunkten und starren Körpern in verschiedenen Koordinatensystemen beschreiben; • können die Bewegungsgleichungen von Massepunkten und starren Körpern mittels der Newtonschen Axiome oder mittels der Lagrangeschen Gleichungen aufstellen; • können die Bewegungsgleichungen für einfache Stoßprobleme lösen; • können die Bewegungsgleichung für einfache Schwingungsprobleme analysieren. 	
7	Voraussetzungen für die Teilnahme	Empfohlen: Kenntnisse aus dem Modul "Statik, Elastostatik und Festigkeitslehre" bzw. "Statik und Festigkeitslehre"	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	NF Solid Mechanics and Dynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 105 h Eigenstudium: 120 h	

14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Gross, Hauger, Schnell, Wall: Technische Mechanik 3, Berlin:Springer, 2006

1	Modulbezeichnung 94580	Statik, Elastostatik und Festigkeitslehre Statics, elastostatics and mechanics of materials	12,5 ECTS
2	Lehrveranstaltungen	Tutorium: Tutorium zur Technischen Mechanik 2 (2 SWS, SoSe 2025) Vorlesung: Technische Mechanik 2 (Elastostatik und Festigkeitslehre) (3 SWS, SoSe 2025) Übung: Übungen zur Technischen Mechanik 2 (UE) (2 SWS, SoSe 2025)	- - -
3	Lehrende	Lucie Spannraff Maximilian Ries Prof. Dr.-Ing. Paul Steinmann	

4	Modulverantwortliche/r	Dr.-Ing. Gunnar Possart Prof. Dr.-Ing. Paul Steinmann	
5	Inhalt	<p>*Statik* (Wintersemester)</p> <ul style="list-style-type: none"> • Kraft- und Momentenbegriff; Axiome der Statik • ebene und räumliche Statik • Flächenmomente 1. und 2. Ordnung • Tribologie • Arbeit/Potential <p>*Elastostatik und Festigkeitslehre* (Sommersemester)</p> <ul style="list-style-type: none"> • Spannung, Formänderung, Stoffgesetz • Zug/Druck-, Biege-, Torsions- und Querschubbeanspruchung schlanker Balken • Energiemethoden der Elastostatik • Elastische Stabilität • Elastizitätstheorie und Festigkeitsnachweis 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • sind vertraut mit den grundlegenden Begriffen und Axiomen der Statik und • können Lager-, Gelenk- und Zwischenreaktionen ebener und räumlicher Tragwerke bestimmen; • erhalten mit den Grundlagen der linearen Thermo-Elastizität (verallgemeinertes Hooke'sches Stoffgesetz) die Befähigung, die Beanspruchung und Deformation in Tragwerken zu ermitteln; • beherrschen die Berechnung der Flächenmomente 1. und 2. Ordnung und • sind befähigt, die Deformationen und Beanspruchungen räumlicher Tragwerke mittels Energiemethoden der Elastostatik (Castigliano/Menabrea) zu bestimmen; • können über Festigkeitshypothesen den Festigkeitsnachweis unter Einbeziehung von Stabilitätskriterien erbringen. 	
7	Voraussetzungen für die Teilnahme	Alle Informationen zum Ablauf der Lehrveranstaltung werden über den StudOn-Kurs kommuniziert. Deshalb bitten wir Sie, sich unter https://www.studon.fau.de/cat5282.html	

		einzuschreiben. Der Beitritt ist nicht, wie sonst üblich, passwortgeschützt, sondern erfolgt nach Bestätigung durch den Dozenten. Dies geschieht mitunter nicht umgehend, aber rechtzeitig vor dem ersten Termin. Wir bitten um Ihr Verständnis.
8	Einpassung in Studienverlaufsplan	Semester: 3;4
9	Verwendbarkeit des Moduls	NF Solid Mechanics and Dynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>Klausur (180 Minuten) Statik, Elastostatik und Festigkeitslehre (Prüfungsnummer: 45801) (englischer Titel: Statics, Elastostatics and Strength of Materials)</p> <p>Prüfungsleistung, Klausur, Dauer (in Minuten): 180, benotet Anteil an der Berechnung der Modulnote: 100.0 %</p> <p>Erstablingung: SS 2023, 1. Wdh.: WS 2023/2024</p>
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 165 h Eigenstudium: 210 h
14	Dauer des Moduls	2 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • Gross, Hauger, Schröder, Wall: Technische Mechanik 1, Berlin:Springer, 2013 • Gross, Hauger, Schröder, Wall: Technische Mechanik 2, Berlin:Springer, 2012

Technische Wahlmodule

1	Modulbezeichnung 42933	Experimental fluid mechanics	5 ECTS
2	Lehrveranstaltungen	Vorlesung mit Übung: Experimental Fluid Mechanics (Strömungsmesstechnik) (3 SWS)	5 ECTS
3	Lehrende	Prof. Dr. Andreas Wierschem	

4	Modulverantwortliche/r	Prof. Dr. Andreas Wierschem	
5	Inhalt	<p>Content:</p> <ul style="list-style-type: none"> • Flow visualization • Measurement techniques for velocity: Particle Image and Tracking Velocimetry and Laser Doppler anemometry, ultrasound, • Measurement techniques for flow rate, pressure, temperature, concentration, free surfaces • Applicability and limitations, typical errors • 2-, 2+1-, 3-dimensional techniques, time-resolved techniques • Data acquisition and processing 	
6	Lernziele und Kompetenzen	<p>Students who participate in this course will become familiar with measurement techniques in fluid mechanics.</p> <p>Students who successfully participate in this module:</p> <ul style="list-style-type: none"> • Have an overview over the most extended and important measurement techniques • Understand the principles of the different techniques • Know and understand the abilities and limitations of the techniques • Can to select an appropriate technique for a given task • Can identify and avoid typical measurement errors 	
7	Voraussetzungen für die Teilnahme	<p>*Prerequisites:*</p> <p>To succeed in this course, students will need to apply acquired knowledge from fluid mechanics. Basic knowledge in physics and measurement techniques is beneficial.</p>	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Variabel mündlich, 30 min	
11	Berechnung der Modulnote	Variabel (100%)	
12	Turnus des Angebots	nur im Sommersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 45 h Eigenstudium: 105 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Englisch	

16	Literaturhinweise	<ul style="list-style-type: none">• Tropea, Yarin, Foss: Handbook of Experimental Fluid Mechanics, Springer• Merzkirch: Flow Visualization, Academic Press• Mayinger, Feldmann: Optical Measurements, Springer
----	--------------------------	--

1	Modulbezeichnung 43385	Physically-based Simulation in Computer Graphics Physically-based simulation in computer graphics	5 ECTS
2	Lehrveranstaltungen	Übung: Tutorials to Physically-based Simulation in Computer Graphics (0 SWS)	2,5 ECTS
		Vorlesung: Physically-based Simulation in Computer Graphics (0 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Tobias Günther	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Tobias Günther	
5	Inhalt	<p>Over the past decades, computer graphics became a vital component of the entertainment industry. Whether in regards to video games, animation movies, or visual effects in live action productions, computer animation brings virtual worlds to life. Thereby, physically-based simulations are required to reach the necessary degree of realism. Based on differential equations and numerical methods to solve them, this lecture will cover a series of algorithms used to implement physically-based simulations. Among others, we are concerned with:</p> <ul style="list-style-type: none"> • kinematics and dynamics of motion (generalized coordinates), • numerical time integration techniques (explicit and implicit time integration), • rigid bodies (simulation, collision detection and response), • deformable objects (mass-spring-systems, finite-elements and thin shells), • grid-based fluid simulation (fractional step method), • particle-based fluid simulation (smoothed particle hydrodynamics and viscosity), • hybrid fluid simulation (fluid implicit particle FLIP, liquid-air interfaces), • adding detail to smoke, fire (vorticity confinement, wavelet turbulence), • shallow water waves and oceans <p>This practical course consists of lectures, programming exercises, and a group programming project.</p>	
6	Lernziele und Kompetenzen	<p>Students learn how to:</p> <ul style="list-style-type: none"> • apply numerical time integration methods at practical examples • derive and analyze the properties of equations of motion • set appropriate boundary conditions • compare numerical solvers regarding stability, accuracy and performance • describe different techniques for rigid body, deformable, and fluid simulations • implement the algorithms in C++ 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 5	

9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Übungsleistung Variabel
11	Berechnung der Modulnote	Übungsleistung (bestanden/nicht bestanden) Variabel (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	

1	Modulbezeichnung 43490	Hardware-Software-Co-Design Hardware-software-co-design	5 ECTS
2	Lehrveranstaltungen	Übung: UE-HSCD (2 SWS) Vorlesung: Hardware-Software-Co-Design (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Muhammad Sabih Tobias Hahn Abrarul Karim Prof. Dr.-Ing. Jürgen Teich	

4	Modulverantwortliche/r	Joachim Falk Prof. Dr.-Ing. Jürgen Teich	
5	Inhalt	<p>Zahlreiche Realisierungen eingebetteter Systeme (z.B. Mobiltelefone, Faxgeräte, Industriesteuerungen) zeichnen sich durch kooperierende Hardware- und Softwarekomponenten aus. Die Popularität solcher Realisierungsformen lässt sich begründen durch 1) die steigende Vielfalt und Komplexität heterogener Systeme, 2) die Notwendigkeit, Entwurfs- und Testkosten zu senken und 3) Fortschritte in Schlüsseltechnologien (Mikroelektronik, formale Entwurfsmethoden). Zum Beispiel bieten Halbleiterhersteller kostengünstige ASICs an, die einen Mikrocontroller und benutzerspezifische Peripherie und Datenpfade auf einem Chip integrieren.</p> <p>Die Synthese solcher Systeme wirft jedoch eine Reihe neuartiger Entwurfsprobleme auf, insbesondere 1) die Frage der Auswahl von Hardware- und Softwarekomponenten, 2) die Partitionierung einer Spezifikation in Hard- und Software, 3) die automatische Synthese von Interface- und Kommunikationsstrukturen und 4) die Verifikation und Cosimulation.</p> <p>1) Überblick und Vergleich von Architekturen und Komponenten in Hardware/Software-Systemen. 2) Aufbau eines Compilers und Codeoptimierungsverfahren für Hardware und Software 3) Hardware/Software-Partitionierung (Partitionierung komplexer Systeme, Schätzungsverfahren, Performanzanalyse, Codegenerierung) 4) Interfacesynthese (Kommunikationsarten, Synchronisation, Synthese) 5) Verifikation und Cosimulation 6) Tafelübungen</p>	
6	Lernziele und Kompetenzen	<p>Fachkompetenz - Wissen</p> <ul style="list-style-type: none"> Die Studierenden erhalten Einblick in ein aktuelles Forschungsgebiet. <p>Fachkompetenz - Verstehen</p> <ul style="list-style-type: none"> Die Studierenden verstehen Grundlagen des modernen Systementwurfs. Die Studierenden erklären Implementierungsalternativen für digitale Hardware/Software-Systeme. <p>Fachkompetenz - Anwenden</p> <ul style="list-style-type: none"> Die Studierenden wenden grundlegende Algorithmen an, zur Analyse und Optimierung von Hardware/Software-Systemen. 	

7	Voraussetzungen für die Teilnahme	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung)“ aus.
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>Empfohlene Bücher zur Begleitung und Vertiefung:</p> <ul style="list-style-type: none"> • Teich J., Haubelt C.: "Digitale Hardware/Software-Systeme: Synthese und Optimierung", Springer-Verlag, 2007, ISBN: 978-3-540-46822-6 • Gajski, D. et al.: "Specification and Design of Embedded Systems", Prentice Hall, Englewood Cliffs, NJ, 1994, ISBN: 978-0131507319 <p>Weitere Informationen:</p> <p>https://www.cs12.tf.fau.de/lehre/lehveranstaltungen/vorlesungen/hardware-software-co-design</p>

1	Modulbezeichnung 43510	Parallele Systeme Parallel systems	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Parallele Systeme (2 SWS) Übung: Übung zu Parallele Systeme (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Teich Frank Hannig Dominik Walter Batuhan Sesli	

4	Modulverantwortliche/r	Joachim Falk Frank Hannig Prof. Dr.-Ing. Jürgen Teich	
5	Inhalt	<p>Aktuelle PCs verfügen über Mehrkernprozessoren und Grafikkarten, die wiederum aus hunderten von einfachen Prozessoren bestehen können. Hierdurch wird ein hohes Maß an nebenläufiger Datenverarbeitung möglich, welche bis vor einigen Jahren nur in Großrechnern erreicht werden konnte. Die effiziente Ausnutzung dieser Parallelität bedarf allerdings mehr als nur mehrerer Prozessoren, insbesondere muss das zu lösende Problem Parallelverarbeitung erlauben. In dieser Vorlesung werden Eigenschaften unterschiedlicher paralleler Rechnerarchitekturen und Metriken zu deren Beurteilung behandelt. Weiterhin werden Modelle und Sprachen zum Programmieren paralleler Rechner eingeführt. Neben der Programmierung von allgemeinen Parallelrechnern werden Entwurfsmethoden (CAD) vorgestellt, wie man ausgehend von einer algorithmischen Problemstellung ein massiv paralleles Rechenfeld in VLSI herleiten kann, das genau dieses Problem optimal parallel berechnet. Solche Schaltungen spielen auf der Bit- bzw. Wortebene eine dominante Rolle (Arithmetik) sowie bei Problemen der Signal- und Bildverarbeitung (z.B. Filter). Schwerpunkt der Vorlesung ist die Vermittlung von Grundlagen der parallelen Datenverarbeitung.</p> <p>Im Einzelnen werden behandelt:</p> <ol style="list-style-type: none"> 1) Theorie der Parallelität (parallele Computermodelle, parallele Spezifikationsformen und -sprachen, Performanzmodelle und -berechnung) 2) Klassifikation paralleler und skalierbarer Rechnerarchitekturen (Multiprozessoren und Multicomputer, Vektorrechner, Datenflussmaschinen, VLSI-Rechenfelder) 3) Programmierbare System-on-Chip (SoC) und Mehrkern-Architekturen (Grafik-Prozessoren, Cell, etc.) 4) Programmierung paralleler Rechner (Sprachen und Modelle, Entwurfsmethoden und Compiler, Optimierung) 5) Massive Parallelität: Vom Algorithmus zur Schaltung <p><i>Today's PCs consist of multi-core processors and graphics cards that again comprise hundreds to thousands of simple processors. As a result of this, a very high degree of parallel data processing becomes possible, which was subjected to supercomputers a couple of years ago. The efficient exploitation of parallel processing requires not only multiple processors but also parallelism inherent in the problem to</i></p>	

		<p>be solved. In this lecture, properties of different parallel computer architectures and corresponding quality metrics are examined. Further, models and parallel programming languages are introduced. In addition to programming general parallel computers, design methods (CAD) are presented that systematically transform an algorithmic problem description into a massive parallel processor array (VLSI), which can optimally execute the given problem in parallel. Such highly parallel circuits play an essential role at the bit level and circuit level (arithmetics) as well as in the case of signal processing and image processing (e.g., filter). The focus of this lecture are foundations of parallel data processing.</p> <p>In detail, the following topics are covered:</p> <ol style="list-style-type: none"> 1) Theory of parallelism (parallel models of computation, parallel specification and parallel languages, performance models) 2) Classification of parallel and scalable computer architectures (multi-processors and multi-computers, vector computers, data-flow machines, VLSI processor arrays) 3) Programmable System-on-Chip (SoC) and multi-core architectures (graphics processors, Cell, etc.) 4) Programming of parallel computers (languages and models, design methods and compiler, optimization) 5) Massive parallelism: From algorithm to circuit
6	<p>Lernziele und Kompetenzen</p>	<p>Schwerpunkt der Vorlesung ist die Vermittlung von Grundlagen der parallelen Datenverarbeitung.</p> <p><i>The focus of this lecture are foundations of parallel data processing.</i></p> <p>Fachkompetenz - Verstehen</p> <ul style="list-style-type: none"> • Die Studierenden verstehen grundlegende Konzepte der parallelen Datenverarbeitung, sowohl theoretischer Art anhand von Modellen, als auch an Architekturbeispielen. The students become familiar with the fundamentals of parallel data processing, theoretic in the form of models as well as by architecture examples. <p>Fachkompetenz - Anwenden</p> <ul style="list-style-type: none"> • Die Studierenden setzen sich mit modernen eingebetteten parallelen Ein-Chip-Architekturen auseinander. The students get familiar with modern embedded parallel system-on-chip architectures. • Die Studierenden wenden grundlegende Performanzmodelle und Parallelisierungstechniken zur Analyse und Optimierung von parallelen Algorithmen und Architekturen an. The students exercise basic performance models and parallelization techniques for the analysis and optimization of parallel algorithms and architectures. • Die Studierenden setzen die Modellierung und den Entwurf von massiv-parallelen Prozessorfeldern in konkreten Aufgaben selbstständig um. In concrete tasks, the students apply independently the modeling and the design of massively parallel processors arrays.

7	Voraussetzungen für die Teilnahme	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Parallele Systeme (Vorlesung mit erweiterten Übungen)“ aus.
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>Klausur (90 Minuten)</p> <ul style="list-style-type: none"> • Ein Wechsel der Prüfungsform von einer Klausur zu einer mündlichen Prüfung ist in Ausnahmefällen (siehe § 16 ABMPO/TechFak) auch nach Semesterbeginn noch möglich. In diesem Fall werden die Studierenden spätestens zwei Wochen nach Vorlesungsbeginn informiert. • Prüfungssprache: Deutsch oder Englisch. Die Unterrichts- und Prüfungssprache hängt von den Sprachkenntnissen und Präferenzen der Teilnehmerinnen und Teilnehmer ab und wird dementsprechend innerhalb der ersten zwei Wochen nach Vorlesungsbeginn festgelegt.
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch oder Englisch
16	Literaturhinweise	<p>Weitere Informationen:</p> <p>https://www.cs12.tf.fau.de/lehre/lehveranstaltungen/vorlesungen/parallele-systeme</p>

1	Modulbezeichnung 43700	Transportprozesse Transport processes	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Dr.-Ing. Sebastian Rieß Prof. Dr.-Ing. Michael Wensing	
5	Inhalt	<ul style="list-style-type: none"> • Transportvorgänge: Wärme-, Stoff-, und Impulsübertragung • Auf Basis der kinetischen Gastheorie werden Gleichungen zur Beschreibung von Transportvorgängen (allgemeine Transportgleichung, Fourier'sches Gesetz, Fick'sche Gesetze,) hergeleitet und für in der Technik typischen Geometrien und Randbedingungen angewandt • Herleitung von Gleichungen zur Beschreibung technischer Aufgabenstellung • Aufbereitung von Problemstellungen zur Lösung mit Rechnerunterstützung 	
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • haben vertiefende Kenntnisse in der Impuls-, Wärme, und Stoffübertragung • können Gleichungen zur Beschreibung technischer Aufgabenstellungen eigenständig herleiten • bereiten Aufgabenstellung zur Lösung am Rechner z.B. mit Hilfe von MatLab auf • erarbeiten projektbezogener Aufgaben am Beispiel von Miniprojekten 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	schriftlich oder mündlich (120 Minuten) variabel: mündlich oder schriftlich	
11	Berechnung der Modulnote	schriftlich oder mündlich (100%)	
12	Turnus des Angebots	nur im Sommersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 45 h Eigenstudium: 105 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Deutsch	

1	Modulbezeichnung 43722	Scientific Visualization Scientific visualization	5 ECTS
2	Lehrveranstaltungen	Übung: Tutorials to Scientific Visualization (2 SWS) Vorlesung: Scientific Visualization (2 SWS)	0 ECTS 5 ECTS
3	Lehrende	Prof. Dr.-Ing. Tobias Günther Xingze Tian	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Tobias Günther	
5	Inhalt	<p>The amount of data, generated in the pursuit of scientific discovery, keeps rapidly increasing across all major scientific disciplines. How can we make sense of large, time-dependent, high-dimensional and multi-variate data? This lecture provides an introduction into scientific visualization. Throughout the course, we cover the fundamental perception basics needed to convey information accurately. After categorizing different data types based on their dimensionality, we dive deeper into specific techniques for scalar, vector and tensor valued data.</p> <p>The lecture covers the following topics:</p> <ul style="list-style-type: none"> • visualization design basics (data abstraction, visual encoding of information), • a review of scalar and vector calculus (differential properties, extremal and critical points), • data structures and data acquisition techniques (grids, interpolation, and differentiation), • indirect volume visualization (marching cubes and contour trees), • direct volume visualization (ray marching and Monte Carlo rendering), • elementary and line-based flow visualization (numerical integration, seeding, rendering), • surface-based flow visualization (integration, selection, rendering), • topology-based flow visualization (topological skeleton, bifurcations, feature flow fields), • feature-based flow visualization (vortices, material boundaries, Lagrangian coherent structures), • advanced methods (tensor visualization, uncertainty, ensembles) 	
6	Lernziele und Kompetenzen	<p>Students are able to:</p> <ul style="list-style-type: none"> • use perception basics to select appropriate visualization methods • classify data and select appropriate visualization techniques • calculate differential properties of scalar and vector fields • identify features in scalar and vector-valued data • implement numerical extraction algorithms • learn the advantages and disadvantages of common visualization techniques 	

7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Variabel Electronic exam in presence with multiple choice questions (90 minutes)
11	Berechnung der Modulnote	Variabel (100%) The final grade of the module is determined by the exam. Exercise bonus: <ul style="list-style-type: none"> • Obtaining more than 80% of the points across all theoretical exercises awards an exam bonus of a third grade.
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	

1	Modulbezeichnung 43950	Kommunikationssysteme Communication systems	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Reinhard German	
5	Inhalt	<p>Aus Rechnerkommunikation ist der grundlegende Aufbau von IP-basierten Netzen bekannt, Inhalt von Kommunikationssysteme sind weitere Netztechnologien wie Leitungsvermittlung (Telefonnetze, Sonet/SDH/WDM) und Netze mit virtueller Leitungsvermittlung (ATM, MPLS) sowie Netzwerkvirtualisierung (SDN, NFV), Multimediakommunikation über paketvermittelte Netze (Streaming, RTP, SIP, Multicast), Dienstgüte in paketvermittelten Netzen (Integrated Services, RSVP, Differentiated Services, Active Queue Management, Policing, Scheduling), drahtlose und mobile Kommunikation (GSM, UMTS, LTE, 5G, Wimax, WLAN, Bluetooth, ZigBee u.a. Sensornetze). Auch Kommunikation in der Industrie wird behandelt. In der Übung werden praktische Aufgaben im Labor durchgeführt: ein Labor enthält mehrere IP-Router, Switches und Rechner, IP-Telefone und Telefonie-Software für VoIP, es werden verschiedene Konfigurationen eingestellt und getestet. Ein weiterer Übungsteil beschäftigt sich mit Mobilkommunikation.</p> <p>*Contents:*</p> <p>Based on the course computer communications the architecture of IP networks is known. Contents of this course will be additional networking technologies such as circuit switching (telephony, SONET/SDH/WDM) and networks with virtual circuit switching (ATM, MPLS) as well as network virtualization (SDN, NFV), multimedia communications over packet switched networks (streaming, RTP, SIP, multicast), quality-of-service in packet switched networks (integrated services, RSVP, differentiated services, active queue management, policing, scheduling), wireless and mobile communications (GSM, UMTS, LTE, 5G, Wimax, WLAN, Bluetooth, sensor networks such as ZigBee). Industrial communication will also be a topic. In the tutorial practical tasks are performed in the laboratory: One laboratory contains several IP routers, switches and computers, IP phones and VoIP telephone software. Various configurations are set up and tested. Another part of the tutorial deals with mobile communications.</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden erlangen Kenntnisse über Technologien bei der Leitungs- und Paketvermittlung in leitungsgebundenen und drahtlosen/mobilen Netzen Kenntnisse über die Grundlagen von Dienstgütemechanismen in paketvermittelten Netzen praktische Erfahrung in der Konfiguration eines IP-Switch-Router-Netzes mit Multimediaverkehr Students obtain the following learning targets and competences</p>	

		<p>Knowledge of technologies in circuit and packet switching in wired and wireless/mobile networks</p> <p>Knowledge of the foundations of quality of service mechanisms in packet switched networks</p> <p>Practical experience in configuring an IP switch router network with multimedia traffic</p>
7	Voraussetzungen für die Teilnahme	Basic knowledge of working with the Linux command line interface (terminal).
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>schriftlich oder mündlich (90 Minuten)</p> <p>Prüfungsleistung, Klausur, Dauer (in Minuten): 90, benotet, 5 ECTS</p> <p>Anteil an der Berechnung der Modulnote: 100.0 %</p> <p>Die Modulprüfung besteht aus:</p> <ul style="list-style-type: none"> • Bearbeitung (zwei)wöchentlicher Aufgabenblätter in Gruppenarbeit. Für diese unbenotete Studienleistung sind alle Aufgabenblätter korrekt zu lösen und abzugeben. • Klausur von 90 Minuten
11	Berechnung der Modulnote	schriftlich oder mündlich (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	<p>Kurose, Ross. Computer Networking: A Top-Down Approach. 7th Ed., Pearson Education, 2017</p> <p>W. Stallings. Data and Computer Communications, 10th ed., Pearson Education, 2014</p> <p>W. Stallings. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, Pearson Education, 2016</p> <p>Cox. An Introduction to LTE. Wiley, 2012</p>

1	Modulbezeichnung 44100	Introduction to the Finite Element Method Introduction to the finite element method	5 ECTS
2	Lehrveranstaltungen	Übung: Introduction to the Finite Element Method - Tutorial (2 SWS) Vorlesung: Introduction to the Finite Element Method (2 SWS)	- -
3	Lehrende	Sebastian Pfaller	

4	Modulverantwortliche/r	Sebastian Pfaller
5	Inhalt	<ul style="list-style-type: none"> • Einführung in die Finite Elemente Methode • Anwendung der Finiten Elemente Methode bei der Modellierung von Stabwerken • Anwendung der Finiten Elemente Methode bei der Modellierung von Balkenstrukturen • Finite Elemente Methode bei Wärmeleitung • Finite Elemente Methode in der Elastizität • Finite Elemente Methode in der Elektrostatik <p>*Contents*</p> <ul style="list-style-type: none"> • Basic concept of the finite element method • Application of the finite element method for the analysis of trusses • Application of the finite element method for the analysis of frames and structures • Finite elements in heat transfer • Finite elements in elasticity • Finite elements in electrostatics
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> - sind vertraut mit der grundlegenden Idee der linearen Finiten Element Methode - können lineare Probleme der Kontinuumsmechanik modellieren - können lineare Wärmeleitungsprobleme modellieren - kennen das isoparametrische Konzept - kennen Verfahren zur numerischen Integration - können ein gegebenes Problem mit Finiten Elementen diskretisieren - können für eine gegebene Differentialgleichung die schwache und diskretisierte Form aufstellen <p>*Objectives*</p> <p>The students</p> <ul style="list-style-type: none"> • are familiar with the basic concept of the finite element method • are able to model linear problems in elasticity • are able to model linear problems in heat transfer • are familiar with the isoparametric concept • know different methods for numerical integration • know how to discretize and solve problems in continuum mechanics • can derive weak and discrete representations of boundary value problems

7	Voraussetzungen für die Teilnahme	<p>Voraussetzungen / Organisatorisches Alle Informationen zum Ablauf der Lehrveranstaltung werden über den StudOn-Kurs kommuniziert. Deshalb bitten wir Sie, sich unter https://www.studon.fau.de/cat5282.html einzuschreiben. Der Beitritt ist nicht, wie sonst üblich, passwortgeschützt, sondern erfolgt nach Bestätigung durch den Dozenten. Dies geschieht mitunter nicht umgehend, aber rechtzeitig vor dem ersten Termin. Wir bitten um Ihr Verständnis.</p> <p>We will communicate all information about the lecture schedule via the StudOn course. Therefore, we ask you to enroll at https://www.studon.fau.de/cat5282.html.</p> <p>The entry is not password-protected, as usual, but takes place after confirmation by the lecturer. The acceptance may not happen immediately, but in time for the first class. We ask for your understanding.</p>				
8	Einpassung in Studienverlaufsplan	Semester: 4				
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010				
10	Studien- und Prüfungsleistungen	<p>Klausur (90 Minuten) Introduction to the Finite Element Method (Prüfungsnummer: 41001) (englischer Titel: Introduction to the Finite Element Method) Prüfungsleistung, Klausur, Dauer (in Minuten): 90, benotet, 5 ECTS Anteil an der Berechnung der Modulnote: 100.0 % Erstablegung: SS 2023, 1. Wdh.: WS 2023/2024</p> <table border="1"> <tr> <td>1. Prüfer:</td> <td>Sebastian Pfaller</td> </tr> </table> <p>Introduction to the Finite Element Method (TAF Solid Mechanics and Dynamics) (Prüfungsnummer: 838659) (englischer Titel: Introduction to the Finite Element Method (TAF Solid Mechanics and Dynamics)) Prüfungsleistung, Klausur, Dauer (in Minuten): 90, benotet, 5 ECTS Anteil an der Berechnung der Modulnote: 100.0 % Erstablegung: SS 2023, 1. Wdh.: WS 2023/2024</p> <table border="1"> <tr> <td>1. Prüfer:</td> <td>Sebastian Pfaller</td> </tr> </table>	1. Prüfer:	Sebastian Pfaller	1. Prüfer:	Sebastian Pfaller
1. Prüfer:	Sebastian Pfaller					
1. Prüfer:	Sebastian Pfaller					
11	Berechnung der Modulnote	Klausur (100%)				
12	Turnus des Angebots	nur im Sommersemester				
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h				
14	Dauer des Moduls	1 Semester				
15	Unterrichts- und Prüfungssprache	Englisch				
16	Literaturhinweise					

1	Modulbezeichnung 44130	Pattern Recognition Pattern recognition	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Andreas Maier	
5	Inhalt	<p>Mathematical foundations of machine learning based on the following classification methods:</p> <ul style="list-style-type: none"> • Bayesian classifier • Logistic Regression • Naive Bayes classifier • Discriminant Analysis • norms and norm dependent linear regression • Rosenblatt's Perceptron • unconstraint and constraint optimization • Support Vector Machines (SVM) • kernel methods • Expectation Maximization (EM) Algorithm and Gaussian Mixture Models (GMMs) • Independent Component Analysis (ICA) • Model Assessment • AdaBoost <p>Mathematische Grundlagen der maschinellen Klassifikation am Beispiel folgender Klassifikatoren:</p> <ul style="list-style-type: none"> • Bayes-Klassifikator • Logistische Regression • Naiver Bayes-Klassifikator • Diskriminanzanalyse • Normen und normabhängige Regression • Rosenblatts Perzeptron • Optimierung ohne und mit Nebenbedingungen • Support Vector Maschines (SVM) • Kernelmethode • Expectation Maximization (EM)-Algorithmus und Gaußsche Mischverteilungen (GMMs) • Analyse durch unabhängige Komponenten • Modellbewertung • AdaBoost 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • verstehen die Struktur von Systemen zur maschinellen Klassifikation einfacher Muster • erläutern die mathematischen Grundlagen ausgewählter maschineller Klassifikatoren • wenden Klassifikatoren zur Lösung konkreter Klassifikationsprobleme an 	

		<ul style="list-style-type: none"> • beurteilen unterschiedliche Klassifikatoren in Bezug auf ihre Eignung • verstehen in der Programmiersprache Python geschriebene Lösungen von Klassifikationsproblemen und Implementierungen von Klassifikatoren <p>Students</p> <ul style="list-style-type: none"> • understand the structure of machine learning systems for simple patterns • explain the mathematical foundations of selected machine learning techniques • apply classification techniques in order to solve given classification tasks • evaluate various classifiers with respect to their suitability to solve the given problem • understand solutions of classification problems and implementations of classifiers written in the programming language Python
7	Voraussetzungen für die Teilnahme	<ul style="list-style-type: none"> • Well grounded in probability calculus, linear algebra/matrix calculus • The attendance of our bachelor course 'Introduction to Pattern Recognition' is not required but certainly helpful. • Gute Kenntnisse in Wahrscheinlichkeitsrechnung und Linearer Algebra/Matrizenrechnung • Der Besuch der Bachelor-Vorlesung 'Introduction to Pattern Recognition' ist zwar keine Voraussetzung, aber sicherlich von Vorteil.
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch oder Englisch Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> • Richard O. Duda, Peter E. Hart, David G. Stock: Pattern Classification, 2nd edition, John Wiley&Sons, New York, 2001 • Trevor Hastie, Robert Tibshirani, Jerome Friedman: The Elements of Statistical Learning - Data Mining, Inference, and Prediction, 2nd edition, Springer, New York, 2009

- Christopher M. Bishop: Pattern Recognition and Machine Learning, Springer, New York, 2006

1	Modulbezeichnung 44460	Architekturen von Superrechnern Architectures of supercomputers	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	
5	Inhalt	<ul style="list-style-type: none"> • Principles of computer and processor architectures • Modern processor architectures • Homogeneous and heterogeneous multi/many-core processors • Parallel computer architectures • Classification and principles of coupling parallel computers • High speed networks in supercomputers • Examples of supercomputers • Programming of supercomputers
6	Lernziele und Kompetenzen	<p>Fachkompetenz Wissen Lernende können die Funktionsweise moderner in Superrechnern eingesetzter Prozessoren wiedergeben. Sie erkennen die besonderen Probleme im Zusammenhang mit dem Energieverbrauch und der Programmierung von Superrechnern.</p> <p>Verstehen Lernende können die Unterschiede bei der Kopplung paralleler Prozesse darstellen. Sie können Parallelrechner hinsichtlich ihrer Speicheranbindung und den grundlegenden Verarbeitungsprinzipien klassifizieren.</p> <p>Anwenden Lernenden sind in der Lage ein eigenes technisches oder mathematisches Problem zur Lösung auf einem Supercomputer umzusetzen und auszuführen. Anhand der in der Vorlesung gezeigten Beispiele sind sie in der Lage, Herausforderungen beim Auffinden von Flaschenhälsen zu verallgemeinern und für ihr konkretes Problem anzuwenden.</p> <p>Analysieren Lernende sind in der Lage, ihre Problemstellungen, z.B. naturwissenschaftliche oder technische Simulationsexperimente, hinsichtlich der Rechen- und Speicheranforderungen für einen Supercomputer geeignet für die Architektur zu charakterisieren.</p> <p>Evaluieren (Beurteilen) Lernende können mithilfe der aufgezeigten Methodiken zur Leistungsmessung von Parallelrechnern unterschiedliche Rechnerarchitekturen bewerten und für ihre Problemstellung die passende Architektur auswählen.</p>
7	Voraussetzungen für die Teilnahme	Keine

8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	schriftlich (30 Minuten)
11	Berechnung der Modulnote	schriftlich (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 120 h Eigenstudium: 30 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	

1	Modulbezeichnung 44470	Cyber-Physical Systems Cyber-physical systems	5 ECTS
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	Modulverantwortliche/r	Joachim Falk Dr.-Ing. Torsten Klie	
5	Inhalt	<p>Klassische Computersysteme zeichnen sich durch eine strikte Trennung von realer und virtueller Welt aus. Moderne Steuerungssysteme, die z.B. in modernen Fahrzeugen verbaut sind und die aus einer Vielzahl von Sensoren und Aktoren bestehen, entsprechen diesem Bild nur sehr eingeschränkt.</p> <p>Diese Systeme, oft "Cyber-Physical Systems (CPS)" genannt, erkennen ihre physische Umgebung, verarbeiten diese Informationen und können die physische Umwelt auch koordiniert beeinflussen. Hierzu ist eine starke Kopplung von physischem Anwendungsmodell und dem Computer-Steuerungsmodell nötig. Im Unterschied zu Eingebetteten Systemen bestehen CPS meist aus vielen vernetzten Komponenten, die sich selbständig untereinander koordinieren.</p> <p>Diese Vorlesung spannt den Bogen von kontrolltheoretischen Grundlagen über Selbstorganisationsprinzipien bis hin zu visionären Anwendungen aus den Bereichen Verkehr und Medizintechnik. Ferner werden Entwurfsmethoden für Cyber-Physical Systems vorgestellt.</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden erläutern, was Cyber-Physical Systems sind und auf welchen technologischen Grundlagen sie aufbauen, insbesondere in den Bereichen Regelungstechnik, Ablaufplanung, Kommunikation und Selbstorganisation bewerten CPS in verschiedenen Anwendungsgebieten</p> <p>stellen den Entwurfsprozess von CPS dar, insbesondere die Modellierung und die grundlegende Programmierung entdecken wesentliche Herausforderungen beim Entwurf, Ausbringung und Einsatz von CPS.</p>	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 3	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	

10	Studien- und Prüfungsleistungen	mündlich (30 Minuten) schriftlich (Klausur, 90 Minuten)
11	Berechnung der Modulnote	mündlich (100%)
12	Turnus des Angebots	in jedem Semester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>Empfohlene Bücher zur Begleitung und Vertiefung:</p> <ul style="list-style-type: none"> • Andrea Bondavalli, Sara Bouchenak und Hermann Kopetz (Hrsg.) Cyber-Physical Systems of Systems: Foundations – A Conceptual Model and Some Derivations: The AMADEOS Legacy. Springer 2016. • Otto Föllinger Regelungstechnik. Hüthig 1992. • Hilmar Jaschek und Holger Voos Grundkurs der Regelungstechnik. Oldenbourg 2010. • Jörg Kahlert Crash-Kurs Regelungstechnik. VDE Verlag 2010. • Peter Marwedel Embedded Systems Design – Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things, 4. Auflage. Springer 2021 • André Platzner Logic Foundations of Cyber-physical Systems. Springer 2018. • Wolfgang Schneider Praktische Regelungstechnik. Vieweg +Teubner 2008. • Walid M. Taha, Abd-Ehamid M. Taha und Johan Thunberg Cyber-physical Systems – A Model-based Approach. Springer 2021. <p>Weitere Informationen:</p> <p>https://www.cs12.tf.fau.de/lehre/lehveranstaltungen/vorlesungen/cyber-physical-systems/</p>

1	Modulbezeichnung 57025	Praktische Softwaretechnik Applied software engineering	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Bernd Hindel Prof. Dr. Detlef Kips Prof. Dr. Dirk Riehle
5	Inhalt	<p>Software ist überall und Software ist komplex. Nicht triviale Software wird von Teams entwickelt. Oft müssen bei der Entwicklung von Softwaresystemen eine Vielzahl von funktionalen und nicht funktionalen Anforderungen berücksichtigt werden. Hierfür ist eine disziplinierte und ingenieurmäßige Vorgehensweise notwendig.</p> <p>Die Vorlesung "Praktische Softwaretechnik" soll ...</p> <ul style="list-style-type: none"> • ein Bewusstsein für die typischen Problemstellungen schaffen, die bei der Durchführung umfangreicher Softwareentwicklungsprojekte auftreten, • ein breites Basiswissen über die Konzepte, Methoden, Notationen und Werkzeuge der modernen Softwaretechnik vermitteln und • die Möglichkeiten und Grenzen ihres Einsatzes im Kontext realistischer Projektumgebungen anhand praktischer Beispiele demonstrieren und bewerten. <p>Die Vorlesung adressiert inhaltlich alle wesentlichen Bereiche der Softwaretechnik. Vorgestellt werden unter anderem</p> <ul style="list-style-type: none"> • traditionelle sowie agile Methoden der Softwareentwicklung, • Methoden der Anforderungsanalyse und des Systementwurfs, • Konzepte der Softwarearchitektur, -implementierung und Dokumentation und • Testen und Qualitätssicherung sowie Prozessverbesserung. <p>Weitere Materialien und Informationen sind hier zu finden:</p> <ul style="list-style-type: none"> • Zeitplan: http://goo.gl/0fy1T • Materialien: Auf StudOn über den Zeitplan <p>Die Teilnahme ist begrenzt. Bitte registrieren Sie sich zeitig für den Kurs auf StudOn, um sicherzustellen, dass Sie einen Platz erhalten.</p>
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • verstehen den Unterschied zwischen "Programmieren im Kleinen" und "Programmieren im Großen" (Softwaretechnik) • wenden grundlegende Methoden der Softwaretechnik über den gesamten Projekt- und Produktlebenszyklus an • kennen die Rolle und Zuständigkeiten der Berufsbilder "Projektleiter", "Anforderungsermittler", "Softwareentwickler" und "Qualitätssicherer"
7	Voraussetzungen für die Teilnahme	Keine

8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	siehe http://goo.gl/JSoUbV

1	Modulbezeichnung 65718	Introduction to Machine Learning Introduction to machine learning	5 ECTS
2	Lehrveranstaltungen	Vorlesung mit Übung: Introduction to Machine Learning (2 SWS) Übung: IntroML-Ex (2 SWS) Übung: IntroML-Tut (2 SWS)	5 ECTS 1,25 ECTS -
3	Lehrende	Dr.-Ing. Vincent Christlein Paul Stöwer	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Andreas Maier	
5	Inhalt	<p>Das Modul hat zum Ziel, die Studierenden mit dem prinzipiellen Aufbau eines Mustererkennungssystems vertraut zu machen. Es werden die einzelnen Schritte von der Aufnahme der Daten bis hin zur Klassifikation von Mustern erläutert. Das Modul beginnt dabei mit einer kurzen Einführung, bei der auch die verwendete Nomenklatur eingeführt wird. Die Analog-Digital-Wandlung wird vorgestellt, wobei der Schwerpunkt auf deren Auswirkungen auf die weitere Signalanalyse liegt. Im Anschluss werden gebräuchliche Methoden der Vorverarbeitung beschrieben. Ein wesentlicher Bestandteil eines Mustererkennungssystems ist die Merkmalsextraktion. Verschiedene Ansätze zur Merkmalsberechnung/-transformation werden gezeigt, darunter Momente, Hauptkomponentenanalyse und Lineare Diskriminanzanalyse. Darüber hinaus werden Möglichkeiten vorgestellt, Merkmalsrepräsentationen direkt aus den Daten zu lernen. Das Modul schließt mit einer Einführung in die maschinelle Klassifikation. In diesem Kontext wird der Bayes- und der Gauss-Klassifikator besprochen.</p> <p>The module aims to familiarize students with the basic structure of a pattern recognition system. The individual steps from the acquisition of data to the classification of patterns are explained. The module starts with a short introduction, which also introduces the used nomenclature. Analog-to-digital conversion is introduced, with emphasis on its impact on further signal analysis. Common methods of preprocessing are then described. An essential component of a pattern recognition system is feature extraction. Various approaches to feature computation/ transformation are demonstrated, including moments, principal component analysis, and linear discriminant analysis. In addition, ways to learn feature representations directly from the data are presented. The module concludes with an introduction to machine classification. In this context, the Bayes and Gauss classifiers are discussed.</p> <p>T</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • erklären die Stufen eines allgemeinen Mustererkennungssystems • verstehen Abtastung, das Abtasttheorem und Quantisierung • verstehen und implementieren Histogrammequalisierung und -dehnung 	

		<ul style="list-style-type: none"> • vergleichen verschiedene Schwellwertmethoden • verstehen lineare, verschiebungsinvariante Filter und Faltung • wenden verschiedene Tief- und Hochpassfilter sowie nichtlineare Filter an • wenden verschiedene Normierungsmethoden an • verstehen den Fluch der Dimensionalität • erklären verschiedene heuristische Merkmalsberechnungsmethoden, z.B. Projektion auf einen orthogonalen Basisraum, geometrische Momente, Merkmale • basierend auf Filterung • verstehen analytische Merkmalsberechnungsmethoden, z.B. Hauptkomponentenanalyse, Lineare Diskriminanzanalyse • verstehen die Basis von Repräsentationslernen • erläutern die Grundlagen der statistischen Klassifikation (Bayes-Klassifikator) • benutzen die Programmiersprache Python, um die vorgestellten Verfahren der Mustererkennung anzuwenden • lernen praktische Anwendungen kennen und wenden die vorgestellten Algorithmen auf konkrete Probleme an <p>The students</p> <ul style="list-style-type: none"> • explain the stages of a general pattern recognition system • understand sampling, the sampling theorem, and quantization • understand and implement histogram equalization and expansion • compare different thresholding methods • understand linear, shift invariant filters and convolution • apply various low-pass, high-pass, and nonlinear filters • apply different normalization methods • understand the curse of dimensionality • explain different heuristic feature calculation methods, e.g. projection on an orthogonal base space, geometric moments, features based on filtering • understand analytical feature computation methods, e.g. principal component analysis, linear discriminant analysis • understand the basis of representation learning • explain the basics of statistical classification (Bayes classifier) • use the programming language Python to apply the presented pattern recognition methods • learn practical applications and apply the presented algorithms to concrete problems
7	Voraussetzungen für die Teilnahme	<p>Ein Mustererkennungssystem besteht aus den folgenden Stufen: Aufnahme von Sensordaten, Vorverarbeitung, Merkmalsextraktion und maschinelle Klassifikation. Dieses Modul beschäftigt sich in erster Linie mit den ersten drei Stufen und schafft damit die Grundlage für weiterführende Module (Pattern Recognition und Pattern Analysis).</p>

		A pattern recognition system consists of the following stages: Sensor Data Acquisition, Preprocessing, Feature Extraction, and Machine Classification. This module primarily deals with the first three stages and thus creates the basis for more advanced modules (Pattern Recognition and Pattern Analysis).
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (60 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> • Vorlesungsfolien/lecture slides • Heinrich Niemann: Klassifikation von Mustern, 2. überarbeitete Auflage, 2003 • Sergios Theodoridis, Konstantinos Koutroumbas: Pattern Recognition, 4. Auflage, Academic Press, Burlington, 2009 • Richard O. Duda, Peter E. Hart, David G. Stock: Pattern Classification, 2. Auflage, John Wiley & Sons, New York, 2001

1	Modulbezeichnung 92290	Kommunikationsnetze Communication networks	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Andre Kaup	
5	Inhalt	<p>*Hierarchische Strukturen von Netzfunktionen* OSI-Schichtenmodell, Kommunikation im OSI-Modell, Datenstrukturen, Vermittlungseinrichtungen</p> <p>* Datenübertragung von Punkt zu Punkt* Signalverarbeitung in der physikalischen Schicht, synchrones und asynchrones Multiplex, Verbindungsarten</p> <p>*Zuverlässige Datenübertragung* Fehlervorwärtskorrektur, Single-Parity-Check-Code, Stop-and-Wait-ARQ, Go-back-N-ARQ, Selective-Repeat-ARQ</p> <p>*Vielfachzugriffsprotokoll* Polling, Token Bus und Token Ring, ALOHA, slotted ALOHA, Carrier-Sensing-Verfahren</p> <p>*Routing* Kommunikationsnetze als Graphen, Fluten, vollständiger Baum und Hamilton-Schleife, Dijkstra-Algorithmus, Bellman-Ford-Algorithmus, statisches Routing mit Alternativen</p> <p>*Warteraumtheorie* Modell und Definitionen, Little's Theorem, Exponentialwarteräume, Exponentialwarteräume mit mehreren Bedienstationen, Halbexponentialwarteräume</p> <p>*Systembeispiel Internet-Protokoll* Internet Protokoll (IP), Transmission Control Protocol (TCP), User Datagram Protocol (UDP)</p> <p>*Multimedianeetze* Klassifikation von multimedialen Anwendungen, Codierung von Multimediadaten, Audio- und Video-Streaming, Protokolle für interaktive Echtzeit-Anwendungen (RTP, RTCP), Dienstklassen und Dienstgütegarantien</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • verstehen den hierarchischen Aufbau von digitalen Kommunikationsnetzen • unterscheiden grundlegende Algorithmen für zuverlässige Datenübertragung mit Rückkanal und beurteilen deren Leistungsfähigkeit • analysieren Protokolle für Vielfachzugriff in digitalen Kommunikationsnetzen und berechnen deren Durchsatz • unterscheiden Routingverfahren und berechnen optimale Vermittlungswege für beispielhafte Kommunikationsnetze 	

		<ul style="list-style-type: none"> • abstrahieren und strukturieren Warteräume in Kommunikationsnetzen und berechnen maßgebliche Kenngrößen wie Aufenthaltsdauer und Belegung • verstehen grundlegende Mechanismen für die verlustlose und verlustbehaftete Codierung von Mediendaten • kennen die maßgeblichen Standards des Internets für Sicherung, Vermittlung und Transport von digitalen Daten
7	Voraussetzungen für die Teilnahme	Kenntnisse über Grundbegriffe der Stochastik
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	M. Bossert, M. Breitbach, "Digitale Netze", Stuttgart: Teubner-Verlag, 1999

1	Modulbezeichnung 92400	Optische Übertragungstechnik Optical communication systems	5 ECTS
2	Lehrveranstaltungen	Übung: Optische Übertragungstechnik Übung (2 SWS) Vorlesung: Optische Übertragungstechnik (2 SWS)	- 5 ECTS
3	Lehrende	Esther Renner Prof. Dr.-Ing. Bernhard Schmauß	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Bernhard Schmauß	
5	Inhalt	<p>Kommerzielle Optische Kommunikationssysteme erreichen pro Faser Übertragungskapazitäten von mehreren Tbit/s. Im Labor wurden mehr als 100Tbit/s nachgewiesen. Die Realisierung derartiger Systeme setzt die Beherrschung verschiedenster Techniken der optischen Übertragungstechnik voraus. In der Vorlesung werden Techniken des Zeitbereichs - (TDM) und Wellenlängenmultiplex (WDM), aber besonders auch der Auslegung der Übertragungsstrecke (Link Design) auf der Basis entsprechender physikalischer und signaltheoretischer Grundlagen behandelt und vertieft. Dabei werden Verfahren besprochen, die sicherstellen, dass sowohl die Signalverzerrungen durch lineare und nichtlineare Fasereffekte als auch die Akkumulation des Verstärkerrauschens begrenzt bleiben. Es wird ausführlich die Systemoptimierung hinsichtlich des optischen Signal-Rausch-Verhältnisses (OSNR) diskutiert sowie auf Techniken des Dispersions- und Nichtlinearitätsmanagements (z.B. Solitonenübertragung) eingegangen. Hierbei wird dem Themenkomplex einer optimalen Streckenauslegung besonders eingehend behandelt. In der Folge werden verschiedene, gebräuchliche Modulationsverfahren einschließlich kohärenter Übertragungsverfahren behandelt, die in neueren Systemen eingesetzt und in experimentellen Systemen getestet werden. Eine Besprechung optischer Verfahren zur Signalregeneration bildet die Brücke zu aktuellen eigenen Forschungsarbeiten.</p> <p>Die vermittelten Grundlagen werden in der Übung zur Vorlesung durch praxisnahe und anschauliche Simulationsbeispiele vertieft.</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • besitzen spezialisiertes und vertieftes Wissen über die Konzeption und Struktur verschiedener optischer Übertragungssysteme. • können die Qualität optischer Datensignale im Kontext verschiedener Systemkonzepte vergleichen und bewerten • sind in der Lage Streckenauslegungen zu entwickeln und zu optimieren. • besitzen methodische Kenntnis zur Bestimmung und Verbesserung der Leistungsfähigkeit optischer Übertragungsstrecken unter Einbeziehung aktueller wissenschaftlicher Ergebnisse. 	
7	Voraussetzungen für die Teilnahme	Komponenten optischer Kommunikationssysteme hilfreich aber nicht obligatorisch	

8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	mündlich (30 Minuten)
11	Berechnung der Modulnote	mündlich (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>Agrawal, G.P.: Fiber-Optic Communication Systems, John Wiley & Sons, 1997</p> <p>Agrawal, G.P.: Nonlinear Fiber Optics, John Wiley & Sons, 3. Auflage, 2001</p> <p>Kaminow, I, Koch, T.: Optical Fiber Telecommunications IVA, Academic Press, 2002</p> <p>Skriptum zur Vorlesung</p> <p>Kaminow, I, Li, T., Willner,A.: Optical Fiber Telecommunications VA, Academic Press, 2008</p>

1	Modulbezeichnung 92410	Komponenten optischer Kommunikationssysteme	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Bernhard Schmauß	
5	Inhalt	Seit Ende der 70er Jahre werden Systeme zur optischen Nachrichtenübertragung eingesetzt. Seither haben sich sowohl deren Übertragungskapazität als auch die Reichweite drastisch erhöht. Die so entstandenen optischen Kommunikationsnetze sind al Rückgrat der weltweiten Kommunikationsinfrastruktur zu sehen. Diese Entwicklungen wurden und werden besonders durch Innovationen auf dem Gebiet der Komponenten und Subsysteme ermöglicht. Im Rahmen der Vorlesung wird auf die physikalischen Grundlagen der wichtigsten Komponenten wie Halbleiterlaser, Modulatoren, Glasfasern, optische Verstärker und Empfangsdioden eingegangen, wobei ein besonderes Augenmerk auf systemrelevante Effekte und Kenngrößen gelegt wird. An Beispielen wird der Einfluss von Komponenteneigenschaften auf die Leistungsmerkmale des Gesamtsystems erläutert. Dabei wird auch auf real eingesetzte oder in Entwicklung befindliche Komponenten und Systeme Bezug genommen.	
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • verstehen den Aufbau und die Funktionsweisen von optoelektronischen und optischen Bauelementen, die in der optischen Übertragungstechnik eingesetzt werden. • können die optischen Eigenschaften der Systemkomponenten und deren Beeinflussung durch die gewählten Betriebsparameter beurteilen. • kennen die verschiedenen Bauelemente und Subsysteme und deren Eigenschaften • können die Bedeutung linearer und nichtlinearer faseroptischer Effekte und deren Auswirkung auf Systemeigenschaften einschätzen. • können faseroptische Übertragungssysteme und ihre komponentenabhängigen Eigenschaften analysieren. • beherrschen den grundlegenden Umgang mit Systemsimulationswerkzeugen zur Dimensionierung faseroptischer Übertragungssysteme. 	
7	Voraussetzungen für die Teilnahme	Empfohlen werden grundlegende Kenntnisse in den Bereichen: <ul style="list-style-type: none"> • Halbleiterphysik • Strahlenoptik • Photonik 	
8	Einpassung in Studienverlaufsplan	Semester: 3	

9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	schriftlich oder mündlich Prüfungsform: mündlich (30 Minuten)
11	Berechnung der Modulnote	schriftlich oder mündlich (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>Agrawal, G.P.: Fiber Optic Communication Systems, Willey, New York, 1992.</p> <p>Voges, E.; Petermann, K.: Optische Kommunikationstechnik, Springer, Berlin, 2002.</p> <p>Kaminow, I, Li, T.: Optical Fiber Telecommunications IVA, Academic Press, 2002.</p> <p>Kaminow, I, Li, T., Willner,A.: Optical Fiber Telecommunications VA, Academic Press, 2008.</p>

1	Modulbezeichnung 92430	Ereignisdiskrete Systeme	5 ECTS
2	Lehrveranstaltungen	Vorlesung mit Übung: Ereignisdiskrete Systeme (4 SWS)	5 ECTS
3	Lehrende	Prof. Dr. Thomas Moor	

4	Modulverantwortliche/r	Prof. Dr. Thomas Moor
5	Inhalt	<p>Formale Sprachen als Modelle ereignisdiskreter Dynamik</p> <ul style="list-style-type: none"> • reguläre Ausdrücke, endliche Automaten, Nerode-Äquivalenz • natürliche Projektion, synchrone Komposition, Konfliktfreiheit. <p>Entwurf ereignisdiskreter Regler:</p> <ul style="list-style-type: none"> • Sicherheitsspezifikation, Konfliktfreiheit • supremale steuerbare Teilsprache, Fixpunktiterationen • Normalität, Regelung unter eingeschränkter Beobachtbarkeit. <p>Anwendungsstudie:</p> <ul style="list-style-type: none"> • Modellbildung eines einfachen technischen Prozesses • Spezifikation/Entwurf/Simulation am Anwendungsbeispiel
6	Lernziele und Kompetenzen	<p>Teilnehmer dieser Veranstaltung</p> <ul style="list-style-type: none"> • erklären, illustrieren und validieren die vorgestellten Grundlagen formaler Sprachen, • entwickeln einfache Ergänzungen zu den vorgestellten Grundlagen formaler Sprachen, • erklären und illustrieren die vorgestellten Entwurfsverfahren, • überprüfen die vorgestellten Entwurfsverfahren hinsichtlich einzelner Lösungseigenschaften, • entwickeln ereignisdiskrete Modelle einfacher technischer Prozesse, einschließlich formaler Spezifikationen, • wählen im Kontext einfacher technischer Prozesse geeignete Entwurfsverfahren aus und wenden diese kritisch an, • bewerten ihre Regelkreise im Simulationsexperiment.
7	Voraussetzungen für die Teilnahme	<p>Es wird empfohlen, eines der folgenden Module zu absolvieren, bevor dieses Modul belegt wird:</p> <ul style="list-style-type: none"> • Regelungstechnik A (Grundlagen) (RT A) • Einführung in die Regelungstechnik (ERT)
8	Einpassung in Studienverlaufsplan	Semester: 6
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Schriftliche Prüfung (Klausur, mit 90 Minuten Dauer).
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h

14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Cassandras, C.G., Lafortune, S.: Introduction to Discrete Event Systems, Kluwer, 1999

1	Modulbezeichnung 92601	Nachrichtentechnische Systeme Communication systems	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Robert Schober Dr.-Ing. Clemens Stierstorfer Prof. Dr. Jörn Thielecke
5	Inhalt	<p>Übertragungstechnik</p> <ul style="list-style-type: none"> • Einführung und Grundbegriffe • Quellensignale und deren Modellierung • Übertragungskanäle und deren Modellierung • Analoge Modulationsverfahren • Pulscodemodulation • Grundbegriffe der Informationstheorie • Digitale Übertragung <p>Systemaspekte</p> <ul style="list-style-type: none"> • Charakterisierung von Übertragungskanälen (Dopplereffekt, Schwundtypen) • wichtige Eigenschaften von Signalen zur Kanalmessung und Datenübertragung (Spreizcodes, Walsh-Folgen, Exponentialfolgen) • Zugriff auf das Übertragungsmedium mittels CDMA, OFDM und CSMA • Anwendung der Verfahren in DRM, UMTS, IEEE 802.11 und GPS als Vertreter typischer Rundfunk-, Mobilfunk-, WLAN- und Mess-Systeme • kurze Einführung in die Verkehrstheorie (Poissonprozess, Durchsatz) • kurze Einführung in Kommunikationsprotokolle, Systemarchitekturen und das Internet-Schichtenmodell.
6	Lernziele und Kompetenzen	<ul style="list-style-type: none"> • Die Studierenden beschreiben die Aufgaben nachrichtentechnischer Systeme. Sie beschreiben und modellieren Signale mathematisch mit Zufallsprozessen und können diese in den Frequenzbereich transformieren. Sie rechnen lineare Größen in logarithmische Darstellungen um (und zurück) und verwenden die Pegelgrößen sicher. • Die Studierenden analysieren analoge Quellensignale, kennen und nutzen dabei die Kenngrößen und Annahmen bzgl. Bandbegrenzung, Spitzenwertbegrenzung usw. Sie unterscheiden analoge und digitale Quellensignale und beschreiben letztere ebenso anhand der üblichen Kenngrößen. • Die Studierenden erläutern die Definition des Übertragungskanals sowie mögliche Ursachen für

Signalverzerrungen und andere Störeinflüsse. Sie beschreiben den Kanal in äquivalenten komplexen Basisband, insbesondere beschreiben und analysieren sie die Ausbreitung von Signalen bei der Funkübertragung sowie auf Kabeln mit den dort auftretenden Effekten (z.B. Mehrwegeausbreitung, Dämpfung usw.). Sie verwenden additives weißes Rauschen zur Modellierung physikalischer Rauschprozesse in Zeit- und Frequenzbereich. Ebenso verwenden und analysieren die Modelle des AWGN-Kanals und des frequenzselektiven Schwundkanals. Sie bewerten Übertragungsverfahren anhand der Kriterien Leistungseffizienz und Bandbreiteneffizienz.

- Die Studierenden analysieren und beschreiben mathematisch die gängigen Amplitudenmodulationsverfahren (Ein- und Zweiseitenbandmodulation, Quadraturamplitudenmodulation) in Zeit- und Frequenzbereich. Dies gilt ebenso für die Frequenzmodulation. Sie bewerten diese Modulationsverfahren im Leistungs-Bandbreiten-Diagramm und analysieren den Einfluss von additiven Störern. Sie beschreiben die Grundstrukturen der zugehörigen Empfänger, insbesondere des Überlagerungsempfängers.
- Die Studierenden beschreiben den Übergang von analogen zu digitalen Signalen und analysieren die Effekte von Abtastung und Quantisierung. Sie untersuchen die Auswirkungen von Kompandierung bei der Quantisierung sowie die Anforderungen an die differentielle Pulscodierung.
- Die Studierenden verwenden das Shannon'sche Informationsmaß, Quellencodierungstheorem und die wechselseitige Information zur mathematischen Beschreibung der Nachrichtenübertragung über gestörte Kanäle. Sie erklären das Kanalcodierungstheorem und analysieren im Detail den AWGN-Kanal und seine Varianten bzgl. informationstheoretische Größen.
- Die Studierenden erklären die digitale Pulsamplitudenmodulation und analysieren die zugehörigen Sender, die Signale sowie die kohärente Demodulation in Zeit- und Frequenzbereich. Sie ermitteln die Fehlerwahrscheinlichkeit und nutzen dazu das Gauß'sches Fehlerintegral und die Error Function. Sie bewerten die digitalen Übertragungsverfahren im Leistungs-Bandbreiten-Diagramm. Die Studierenden verstehen die Motivation für den Einsatz von Kanalcodierung bei digitaler Übertragung.
- - Grundlegende Methoden und Signale zur Kanalmessung und zum Kanalzugriff
 - Grundlegendes zu Strukturen und Protokollen in Kommunikationssystemen
- Die Studierenden lernen nachrichtentechnischen Signale und Verfahren anzuwenden und zu analysieren.

7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (120 Minuten)
11	Berechnung der Modulnote	<p>Klausur (100%)</p> <p>Hausaufgaben/Bonuspunkte</p> <ul style="list-style-type: none"> • Es können durch das Lösen von Hausaufgaben während des Semsters bis zu 12 Bonuspunkte erworben werden. Diese werden bei bestandener Prüfung zusätzlich in die Bewertung mit einbezogen.
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • Skripten zu den Vorlesungen • Kammeyer: Nachrichtenübertragung, Teubner Verlag, 3. Aufl. • Anderson, Johannesson: Understanding Information Transmission, John Wiley, 2005

1	Modulbezeichnung 93015	Einführung in die moderne Kryptographie Introduction to modern cryptography	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Carina Harrius Prof. Dr. Dominique Schröder	
5	Inhalt	<p>This course gives a comprehensive introduction to modern cryptography. The course also serves as a base for other courses on cryptography that are offered by the chair. The topics covered are the following:</p> <ul style="list-style-type: none"> • Information theoretic security • Computational security • Private key Encryption • Message Authentication Codes • Hash functions • Public key Encryption • Digital Signatures <p>More advanced topics may be covered if time permits.</p>	
6	Lernziele und Kompetenzen	On successfully passing the course, the student is guaranteed to be knowledgeable on the basic concepts of provable security.	
7	Voraussetzungen für die Teilnahme	No previous knowledge in Cryptography or computer Security is required.	
8	Einpassung in Studienverlaufsplan	Semester: 3	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Variabel	
11	Berechnung der Modulnote	Variabel (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 165 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Englisch	
16	Literaturhinweise	<p>Introduction to Modern Cryptography</p> <p>Jonathan Katz and Yehuda Lindell 2nd Edition (2014)</p> <p>(Chapman & Hall/CRC Cryptography and Network Security Series)</p>	

ISBN-13: 978-1466570269

1	Modulbezeichnung 93040	Parallele und Funktionale Programmierung Parallel and functional programming	5 ECTS
2	Lehrveranstaltungen	Übung: Intensivübungen zu Parallele und Funktionale Programmierung (2 SWS)	0 ECTS
3	Lehrende	David Schwarzbeck Dr.-Ing. Norbert Oster Prof. Dr. Michael Philippsen	

4	Modulverantwortliche/r	Dr.-Ing. Norbert Oster Prof. Dr. Michael Philippsen	
5	Inhalt	<ul style="list-style-type: none"> • Grundlagen der funktionale Programmierung • Grundlagen der parallelen Programmierung • Datenstrukturen • Objektorientierung • Scala-Kenntnisse • Erweiterte JAVA-Kenntnisse • Aufwandsabschätzungen • Grundlegende Algorithmen 	
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • erlernen die Grundlagen der funktionalen Programmierung anhand der Programmiersprache Scala • verstehen paralleles Programmieren mit Java • kennen fundamentale Datenstrukturen und Algorithmen • können funktionale und parallele Algorithmen entwickeln und analysieren 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 3	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (60 Minuten)	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Deutsch	
16	Literaturhinweise		

1	Modulbezeichnung 93150	Rechnerkommunikation Computer communications	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Rechnerkommunikation (2 SWS) Übung: Übungen Rechnerkommunikation (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Reinhard German Dr.-Ing. Peter Bazan Mamdouh Muhammad	

4	Modulverantwortliche/r	Prof. Dr. Reinhard German	
5	Inhalt	<p>Die Vorlesung vermittelt die Grundlagen der Rechnerkommunikation und durchläuft von oben nach unten die Schichten des Internets:</p> <ul style="list-style-type: none"> • Anwendungsschicht • Transportschicht • Netzwerkschicht • Sicherungsschicht • Physikalische Schicht <p>Sicherheit wird als übergreifender Aspekt behandelt. An verschiedenen Stellen werden analytische Modelle eingesetzt, um Wege für eine quantitative Auslegung von Kommunikationsnetzen aufzuzeigen. Die Übung beinhaltet praktische und theoretische Aufgaben zum Verständnis der einzelnen Schichten.</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden erwerben</p> <ul style="list-style-type: none"> • Kenntnisse über zentrale Mechanismen, Protokolle und Architekturen der Rechnerkommunikation (Topologie, Schicht, Adressierung, Wegsuche, Weiterleitung, Flusskontrolle, Überlastkontrolle, Fehlersicherung, Medienzugriff, Bitübertragung) am Beispiel des Internets und mit Ausblicken auf andere Netztechnologien • Kenntnisse über Sicherheit, Leistung und Zuverlässigkeit bei der Rechnerkommunikation • praktische Erfahrung in der Benutzung und Programmierung von Rechnernetzen 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	<p>Übungsleistung Klausur (90 Minuten) Hausaufgaben zu Rechnerkommunikation (Übungsleistung):</p> <ul style="list-style-type: none"> • Studienleistung, Übungsleistung, unbenotet, 2.5 ECTS • weitere Erläuterungen: Bearbeitung (zwei)wöchentlicher Aufgabenblätter in Gruppenarbeit. Für den unbenoteten Übungsschein sind 60% der Punkte je Aufgabenblatt zu erreichen <p>Rechnerkommunikation (Klausur):</p>	

		<ul style="list-style-type: none"> • Prüfungsleistung, Klausur, Dauer (in Minuten): 90, benotet, 2.5 ECTS • Anteil an der Berechnung der Modulnote: 100.0 %
11	Berechnung der Modulnote	Übungsleistung (bestanden/nicht bestanden) Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Lehrbuch: Kurose, Ross. Computer Networking. 8th Ed., Pearson, 2021.

1	Modulbezeichnung 93175	Visualization	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Tobias Günther	
5	Inhalt	<p>An old English adage says "a picture is worth a 1,000 words", meaning that complex ideas are often easier to convey visually. This lecture is about the craft of creating informative images from data. Starting from the basics of the human visual perception, we will learn how visualizations are designed for explorative, communicative or confirmative purposes. We will see how data can be classified, allowing us to develop algorithms that apply to a wide range of application domains.</p> <p>The lecture covers the following topics:</p> <ul style="list-style-type: none"> • data abstraction (data types, data set types, attribute types), • perception and mapping (marks and channels, effectiveness, pre- attentive vision, color maps), • task abstraction and validation (actions and targets), • information visualization tools (HTML, CSS, JavaScript, React, D3), • information visualization methods (tabular data, networks, trees), • scientific visualization methods (volume rendering and particle visualization), • scientific visualization tools (VTK, ParaView), • view manipulation (navigation, selection, multiple views), • data reduction (filtering, aggregation, focus and context), • lies in visualization (human biases and rules of thumb), • applications (deep learning, medical visualization, optimization) <p>The lecture is accompanied by exercises. Theoretical exercises concentrate on the classification of data and the design and analysis of visualizations, while programming exercises using web-based technologies give examples of their implementation.</p>	
6	Lernziele und Kompetenzen	<p>Students learn to:</p> <ul style="list-style-type: none"> • use perceptual basics to select appropriate visualization methods • explain the steps of the visualization pipeline • calculate direct and indirect volume visualizations to given data • explain and apply interaction concepts • perform a data and requirement analysis for a given problem • explain visualization techniques for scientific and abstract data 	

7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Variabel The module is concluded with a written exam (90 minutes).
11	Berechnung der Modulnote	Variabel (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	Visualization Analysis and Design, Tamara Munzner, 2014.

1	Modulbezeichnung 93330	Deep Learning for Beginners Deep learning for beginners	2,5 ECTS
2	Lehrveranstaltungen	Vorlesung: Deep Learning for Beginners (VHB-Kurs) (2 SWS)	2,5 ECTS
3	Lehrende	Aline Sindel	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Andreas Maier
5	Inhalt	<p>Neural networks have had an enormous impact on research in image and signal processing in recent years. In this course, you will learn all the basics about deep learning in order to understand how neural network systems are built. The course is addressed to students who are new to the field. In the beginning of the course, we introduce you to the topic with some applications of deep learning in the field of medical imaging, digital humanities and industry projects. Before we dive into the core elements of neural networks, there are two lecture units on the fundamentals of signal and image processing to teach you relevant parts of system theory such as convolutions, Fourier transform, and sampling theorem. In the next lecture units, you learn the basic blocks of neural networks, such as backpropagation, fully connected layers, convolutional layers, activation functions, loss functions, optimization, and regularization strategies. Then, we look into common practices for training and evaluating neural networks. The next lecture unit is focusing on common neural network architectures, such as LeNet, Alexnet, and VGG. It follows a lecture unit about unsupervised learning that contains the principles of autoencoders and generative adversarial networks. Lastly, we cover some applications of deep learning in segmentation and object detection.</p> <p>The accompanying programming exercises will provide a deeper understanding of the workings and architecture of neural networks, in which you will develop a basic neural network from scratch in pure Python without using deep learning frameworks, such as PyTorch or TensorFlow.</p> <p>At the end of the semester, there will be a written exam.</p>
6	Lernziele und Kompetenzen	<p>The students</p> <ul style="list-style-type: none"> • explain the different neural network components, • compare and analyze methods for optimization and regularization of neural networks, • compare and analyze different CNN architectures, • explain deep learning techniques for unsupervised / semi-supervised and weakly supervised learning, • explain different deep learning applications, • implement the presented methods in Python, • effectively investigate raw data, intermediate results and results of Deep Learning techniques on a computer, • autonomously supplement the mathematical foundations of the presented methods by self-guided study of the literature, • discuss the social impact of applications of deep learning applications.

7	Voraussetzungen für die Teilnahme	Requirements: mathematics for engineering, basic knowledge of python
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Variabel (60 Minuten) The final exam is a written exam with 60 minutes duration.
11	Berechnung der Modulnote	Variabel (100%)
12	Turnus des Angebots	in jedem Semester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 0 h Eigenstudium: 75 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	

1	Modulbezeichnung 93510	Digitale Übertragung Digital communications	5 ECTS
2	Lehrveranstaltungen	Übung: Digitalen Übertragung - Übungen (1 SWS) Vorlesung: Digitale Übertragung (3 SWS)	- 5 ECTS
3	Lehrende	Bastian Heinlein Prof. Dr.-Ing. Robert Schober	

4	Modulverantwortliche/r	Prof. Dr. Laura Cottatellucci Prof. Dr.-Ing. Robert Schober Dr.-Ing. Clemens Stierstorfer
5	Inhalt	Alle modernen Kommunikationssysteme basieren auf digitalen Übertragungsverfahren. Das Modul befasst sich mit den Grundlagen der Analyse und des Entwurfs digitaler Sender und Empfänger. Dabei wird zunächst von einem einfachen Kanalmodell bei dem das Empfangssignal nur durch additives weißes Gaußsches Rauschen gestört wird ausgegangen. Im Verlauf werden aber auch Kanäle mit unbekannter Phase sowie verzerrende Kanäle betrachtet. Behandelt werden unter anderem digitale Modulationsverfahren (z.B. Pulsamplitudenmodulation (PAM), digitale Frequenzmodulation (FSK), und Kontinuierliche-Phasenmodulation (CPM)), Orthogonalkonstellationen, das Nyquistkriterium in Zeit- und Frequenzbereich, optimale kohärente und inkohärente Detektions- und Decodierungsverfahren, die Signalraumdarstellung digital modulierter Signale, verschiedene Entzerrungsverfahren, und Mehrträger-Übertragungsverfahren.
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • analysieren und klassifizieren digitale Modulationsverfahren hinsichtlich ihrer Leistungs- und Bandbreiteneffizienz sowie ihres Spitzenwertfaktors, • ermitteln notwendige Kriterien für impulsinterferenzfreie Übertragung, • charakterisieren digitale Modulationsverfahren im Signalraum, • ermitteln informationsverlustfreie Demodulationsverfahren, • entwerfen optimale kohärente und inkohärente Detektions- und Decodierungsverfahren, • vergleichen verschiedene Entzerrungsverfahren hinsichtlich deren Leistungsfähigkeit und Komplexität, • entwerfen einfache digitale Übertragungssysteme mit vorgeschriebenen Leistungs- und Bandbreiteneffizienzen sowie Spitzenwertfaktoren.
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010

10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 93580	Stochastische Prozesse Stochastic processes	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Stochastic Processes (3 SWS) Übung: Ergänzungen und Übungen zu Stochastische Prozesse (1 SWS)	5 ECTS -
3	Lehrende	Prof. Dr.-Ing. Sebastian Schlecht	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Walter Kellermann	
5	Inhalt	<p>*Wahrscheinlichkeitsrechnung und Zufallsvariablen*</p> <p>Wahrscheinlichkeit, Zufallsvariablen, uni- und multivariate Wahrscheinlichkeitsverteilungen und dichten; Funktionen von Zufallsvariablen und deren Verteilungen und dichten; Erwartungswerte; spezielle Verteilungen (diskrete und kontinuierliche); Grenzwertsätze</p> <p>*Stochastische Prozesse*</p> <p>Verteilungen, Dichten und Erwartungswerte eindimensionaler Stochastischer Prozesse; Stationarität, Zyklstationarität, Ergodizität; Schwach stationäre, zeitkontinuierliche und zeitdiskrete Prozesse im Zeit- und Frequenzbereich; lineare zeitinvariante (LZI) Systeme und schwach stationäre Prozesse</p> <p>*Schätztheorie*</p> <p>Punkt- und Intervallschätzung; Schätzkriterien; Prädiktion; klassische und Bayessche Parameterschätzung (inkl. MMSE, Maximum Likelihood, Maximum A Posteriori); Cramer-Rao-Schranke; Hypothesentests und Entscheidungsverfahren (binäre Entscheidungen, Teststatistiken, Chi-Quadrat-Test); Binäre Entscheidungen, Neyman-Pearson-Kriterium</p> <p>*Lineare Optimalfilterung*</p> <p>Orthogonalitätsprinzip; zeitkontinuierliche und zeitdiskrete Wiener-Filterung; adaptive Filter (LMS, NLMS); zeitkontinuierliche und zeitdiskrete Signalangepasste Filter</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • analysieren die statistischen Eigenschaften von Zufallsvariablen und Stochastischen Prozessen mittels Wahrscheinlichkeitsdichten und Erwartungswerten • verstehen die Unterschiede zwischen allgemeinen, stationären und ergodischen Prozessen • kennen die spezielle Rolle der Gaußverteilung und ihre Auswirkungen auf die Eigenschaften von Zufallsvariablen und Prozesse • analysieren die statistischen Eigenschaften von Zufallsprozessen am Ausgang von LZI-Systemen im Zeitbereich und im Frequenzbereich • verstehen die Unterschiede klassischer und Bayesscher Schätzung, entwerfen und analysieren MMSE- und ML-Schätzer für spezielle Schätzprobleme • kennen elementare Hypothesentests und Entscheidungsverfahren • analysieren Optimalfilterprobleme und wenden das Orthogonalitätsprinzip zur Ableitung optimaler Filter an 	

		<ul style="list-style-type: none"> verstehen und wenden das Konzept der signalangepassten Filterung an
7	Voraussetzungen für die Teilnahme	Vorlesung Signale und Systeme I & II
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Hänsler: Statistische Signale, Springer 1998; Papoulis/Pillai: Probability, Random Variables, and Stochastic Processes, Prentice Hall, 2002

1	Modulbezeichnung 94550	Methode der Finiten Elemente Finite element methods	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Methode der Finiten Elemente (2 SWS) Tutorium: Tutorium zur Methode der Finiten Elemente (0 SWS) Übung: Übungen zur Methode der Finiten Elemente (2 SWS)	- - -
3	Lehrende	Prof. Dr.-Ing. Kai Willner Dr.-Ing. Gunnar Possart	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Kai Willner
5	Inhalt	<p>Modellbildung und Simulation</p> <p>Mechanische und mathematische Grundlagen</p> <ul style="list-style-type: none"> • Das Prinzip der virtuellen Verschiebungen • Die Methode der gewichteten Residuen <p>Allgemeine Formulierung der FEM</p> <ul style="list-style-type: none"> • Formfunktionen • Elemente für Stab- und Balkenprobleme • Locking-Effekte • Isoparametrisches Konzept • Scheiben- und Volumenelemente <p>Numerische Umsetzung</p> <ul style="list-style-type: none"> • Numerische Quadratur • Assemblierung und Einbau von Randbedingungen • Lösen des linearen Gleichungssystems • Lösen des Eigenwertproblems • Zeitschrittintegration
6	Lernziele und Kompetenzen	<p>Wissen</p> <ul style="list-style-type: none"> • Die Studierenden kennen verschiedene Diskretisierungsverfahren zur Behandlung kontinuierlicher Systeme. • Die Studierenden kennen das prinzipielle Vorgehen bei der Diskretisierung eines mechanischen Problems mit der Methode der finiten Elementen und die entsprechenden Fachtermini wie Knoten, Elemente, Freiheitsgrade etc. • Die Studierenden kennen die Verschiebungsdifferentialgleichungen für verschiedene Strukturelemente wie Stäbe, Balken, Scheiben und das 3D-Kontinuum. • Die Studierenden kennen die Methode der gewichteten Residuen in verschiedenen Varianten. • Die Studierenden kennen das Prinzip der virtuellen Arbeiten in den verschiedenen Ausprägungen fuer Stäbe, Balken, Scheiben und das 3D-Kontinuum. • Die Studierenden kennen verschiedene Randbedingungstypen und ihre Behandlung im Rahmen der Methode der gewichteten Residuen bzw. des Prinzips der virtuellen Verschiebungen.

		<ul style="list-style-type: none"> • Die Studierenden kennen die Anforderungen an die Ansatz- und Wichtungsfunktionen und können die gängigen Formfunktionen für verschiedene Elementtypen angeben. • Die Studierenden kennen das isoparametrische Konzept. • Die Studierenden kennen Verfahren zur numerischen Quadratur. • Die Studierenden kennen Verfahren zur Lösung linearer Gleichungssysteme, zur Lösung von Eigenwertproblemen und zur numerischen Zeitschrittintegration. <p>Verstehen</p> <ul style="list-style-type: none"> • Die Studierenden verstehen den Zusammenhang zwischen der Methode der gewichteten Residuen und dem Prinzip der virtuellen Arbeiten bei mechanischen Problemen. • Die Studierenden verstehen den Unterschied zwischen schubstarrer und schubweicher Balkentheorie sowie die daraus resultierenden unterschiedlichen Anforderungen an die Ansatzfunktionen. • Die Studierenden verstehen das Problem der Schubversteifung. • Die Studierenden können das isoparametrische Konzept erläutern, die daraus resultierende Notwendigkeit numerischer Quadraturverfahren zur Integration der Elementmatrizen und das Konzept der zuverlässigen Integration erklären. • Die Studierenden können den Unterschied zwischen Lagrange- und Serendipity-Elementen sowie die jeweiligen Vor- und Nachteile erläutern. <p>Anwenden</p> <ul style="list-style-type: none"> • Die Studierenden können ein gegebenes Problem geeignet diskretisieren, die notwendigen Indextafeln aufstellen und die Elementmatrizen zu Systemmatrizen assemblieren. • Die Studierenden können die Randbedingungen eintragen und das Gesamtsystem entsprechend partitionieren. • Die Studierenden können polynomiale Formfunktionen vom Lagrange-, Serendipity- und Hermite-Typ konstruieren. • Die Studierenden können für die bekannten Elementtypen die Elementmatrizen auf analytischen bzw. numerischen Weg berechnen. <p>Analysieren</p> <ul style="list-style-type: none"> • Die Studierenden können für eine gegebene, lineare Differentialgleichung die schwache Form aufstellen, geeignete Formfunktionen auswählen und eine entsprechende Finite-Elemente-Formulierung aufstellen.
7	<p>Voraussetzungen für die Teilnahme</p>	<p>Alle Informationen zum Ablauf der Lehrveranstaltung werden über den StudOn-Kurs kommuniziert. Deshalb bitten wir Sie, sich unter https://www.studon.fau.de/cat5282.html einzuschreiben. Der Beitritt ist nicht, wie sonst üblich, passwortgeschützt, sondern erfolgt nach Bestätigung durch den Dozenten. Dies geschieht mitunter nicht umgehend, aber rechtzeitig vor dem ersten Termin. Wir bitten um Ihr Verständnis.</p>

8	Einpassung in Studienverlaufsplan	Semester: 6
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (60 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 60 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • Knothe, Wessels: Finite Elemente, Berlin:Springer • Hughes: The Finite Element Method, Mineola:Dover

1	Modulbezeichnung 94570	Produktionstechnik I und II Production engineering I+II	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Produktionstechnik (PT II & PT 3 MB) (2 SWS, SoSe 2025) Tutorium: Produktionstechnik - Tutorium (P) (PT II & PT 3 MB) (2 SWS, SoSe 2025)	- -
3	Lehrende	Simon Sauer Andreas Röckelein Prof. Dr. Nico Hanenkamp Prof. Dr.-Ing. Jörg Franke Prof. Dr.-Ing. Dietmar Drummer Prof. Dr.-Ing. Sebastian Müller	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Dietmar Drummer Prof. Dr.-Ing. Jörg Franke Prof. Dr. Nico Hanenkamp Prof. Dr.-Ing. Marion Merklein Prof. Dr.-Ing. Sebastian Müller Prof. Dr.-Ing. Michael Schmidt
5	Inhalt	<p>*Produktionstechnik I:*</p> <p>Basierend auf der DIN 8580 werden die aktuellen Technologien sowie die dabei eingesetzten Maschinen in den Bereichen Urformen, Umformen, Trennen, Fügen, Beschichten und das Ändern der Stoffeigenschaften behandelt. Hierbei werden sowohl die Prozessketten als auch die spezifischen Eigenschaften der Produktionstechniken aufgezeigt und anhand von praxisrelevanten Bauteilen erläutert. Zum besseren Verständnis der Verfahren werden zunächst metallkundliche Grundlagen, wie der mikrostrukturelle Aufbau von metallischen Werkstoffen und ihr plastisches Verhalten, erläutert. Im weiteren Verlauf erfolgt eine Gegenüberstellung der Verfahren der Massivumformung Stauchen, Schmieden, Fließpressen und Walzen. Im Rahmen des Kapitels Blechumformung wird die Herstellung von Bauteilen durch Tiefziehen, Streckziehen und Biegen betrachtet. Der Fokus in der Vorstellung der Verfahrensgruppe Trennen liegt auf den Prozessen des Zerteilens und Spanens. Der Bereich Fügen behandelt die Herstellung von Verbindungen mittels Umformen, Schweißen und Löten. Abschließend werden verschiedene strahlbasierte Fertigungsverfahren aus den sechs Bereichen vorgestellt. Im Fokus stehen hierbei laserbasierte Fertigungsverfahren, wie zum Beispiel Schweißen, Schneiden oder Additiven Fertigung. Eine zusätzlich angebotene Übung dient der Vertiefung und der Anwendung des Vorlesungsinhaltes. Außerdem wird die Verarbeitung von Kunststoffen (Spritzgießen, Erzeugung von duroplastischen / thermoplastischen Faserverbunden) behandelt.</p> <p>*Produktionstechnik II:*</p> <p>Des Weiteren werden die Grundlagen zu Werkzeugmaschinen und dem Werkzeugmaschinenbau (Maschinenkomponenten, Funktionalitäten, Anwendungs- / Einsatzmöglichkeiten) sowie zu</p>

		Montagetechnologien und Verbindungstechniken (Auslegung von Verbindungen, prozesstechnische Umsetzung und Realisierung) vermittelt. Einen weiteren Schwerpunkt stellen der Elektromaschinenbau und die Elektronikproduktion (Funktionsweise und Herstellung von elektronischen Antriebseinheiten, Auslegung und Herstellung von elektronischen Komponenten) dar. Anschließend werden die Urformverfahren Gießen und Pulvermetallurgie dargestellt.
6	Lernziele und Kompetenzen	<p>Fachkompetenz</p> <p>Wissen</p> <ul style="list-style-type: none"> • Die Studierenden erwerben grundlegende Kenntnisse in der Metallkunde und der Verarbeitung von Metallen. • Die Studierenden erhalten einen Überblick über die Produktionsverfahren Urformen, Umformen, Fügen, Trennen, ihre Untergruppen • Die Studierenden erwerben ein grundlegendes Prozessverständnis hinsichtlich der wirkenden Mechanismen. • Die Studierenden erwerben Wissen über die Prozessführung sowie spezifische Eigenschaften der Produktionsverfahren. • Die Studierenden erwerben grundlegendes Verständnis zu den Eigenschaften von Kunststoffen und deren Verarbeitung • Die Studierenden erwerben Kenntnisse über werkstoffwissenschaftliche Aspekte und Werkstoffeigenschaften sowie Werkstoffverhalten vor und nach den jeweiligen Bearbeitungsprozessen • Die Studierenden erwerben fundamentale Kenntnisse zu Multi-Materialien-Verbunden. • Die Studierenden erlangen grundlegende Kenntnisse zur Funktionsweise von elektrischen Antriebseinheiten und deren Herstellung sowie die Herstellung von elektrischen Komponenten (MID) • Die Studierenden erhalten grundlegende Kenntnisse im Bereich der Produktentwicklung und Produktauslegung (Verfahrensmöglichkeiten, Verfahrensgrenzen, Designeinschränkungen, etc.) <p>Verstehen</p> <ul style="list-style-type: none"> • Die Studierenden sind in der Lage die grundlegenden Prinzipien von Fertigungsprozessen und der Systemauslegung zu verstehen • Die Studierenden verstehen die Grundlagen des Anlagen- und Werkzeugbaus <p>Anwenden</p> <ul style="list-style-type: none"> • <ul style="list-style-type: none"> ◦ Die Studierenden können die verschiedenen Fertigungsverfahren erkennen und normgerecht differenzieren
7	Voraussetzungen für die Teilnahme	Keine

8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (120 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	2 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 94660	Statik und Festigkeitslehre Statics and mechanics of materials	7,5 ECTS
2	Lehrveranstaltungen	Vorlesung: Statik und Festigkeitslehre (3 SWS) Tutorium: Statik und Festigkeitslehre (Tut) (2 SWS) Übung: Statik und Festigkeitslehre (Ü) (2 SWS)	- - -
3	Lehrende	Prof. Dr.-Ing. Sigrid Leyendecker Gamal Amer Dr.-Ing. Xiyu Chen	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Sigrid Leyendecker Prof. Dr.-Ing. Kai Willner
5	Inhalt	<ul style="list-style-type: none"> • Kraft- und Momentenbegriff, Axiome der Statik • ebene und räumliche Statik • Flächenmomente 1. und 2. Ordnung • Haft- und Gleitreibung • Spannung, Formänderung, Stoffgesetz • überbestimmte Stabwerke, Balkenbiegung • Torsion • Elastizitätstheorie und Festigkeitsnachweis • Stabilität
6	Lernziele und Kompetenzen	<p>Wissen Die Studierenden kennen</p> <ul style="list-style-type: none"> • die axiomatischen Grundlagen der Technischen Mechanik sowie die entsprechenden Fachtermini. • das Schnittprinzip und die Einteilung der Kräfte in eingeprägte und Reaktionskräfte bzw. in äußere und innere Kräfte. • die Gleichgewichtsbedingungen am starren Körper. • das Phänomen der Haft- und Gleitreibung. • die Begriffe der Verzerrung und Spannung sowie das linear-elastische Stoffgesetz. • den Begriff der Hauptspannungen sowie das Konzept der Vergleichsspannung und Festigkeitshypothesen. • das Problem der Stabilität und speziell die vier Eulerschen Knickfälle für ein schlankes Bauteil unter Drucklast. <p>Verstehen Die Studierenden</p> <ul style="list-style-type: none"> • können Kräfte nach verschiedenen Kriterien klassifizieren. • können verschiedene Lagerungsarten unterscheiden und die entsprechenden Lagerreaktionen angeben. • können den Unterschied zwischen statisch bestimmten und unbestimmten Systemen erklären. • können den Unterschied zwischen Haft- und Gleitreibung erläutern. • können das linear-elastische, isotrope Materialgesetz angeben und die Bedeutung der Konstanten erläutern. • können die Voraussetzungen der Euler-Bernoulli-Theorie schlanker Balken erklären.

		<ul style="list-style-type: none"> verstehen die Idee der Vergleichsspannung und können verschiedene Festigkeitshypothesen erklären. <p>Anwenden</p> <p>Die Studierenden können</p> <ul style="list-style-type: none"> den Schwerpunkt eines Körpers bestimmen. ein System aus mehreren Körpern geeignet freischneiden und die entsprechenden eingprägten Kraftgrößen und die Reaktionsgrößen eintragen. für ein statisch bestimmtes System die Reaktionsgrößen aus den Gleichgewichtsbedingungen ermitteln. die Schnittreaktionen für Stäbe und Balken bestimmen. die Spannungen im Querschnitt schlanker Bauteile (Stab, Balken) unter verschiedenen Belastungen (Zug, Biegung, Torsion) ermitteln. die Verformungen schlanker Bauteile ermitteln. aus einem gegebenen, allgemeinen Spannungszustand die Hauptspannungen sowie verschiedene Vergleichsspannungen ermitteln. die kritische Knicklast für einen gegebenen Knickfall bestimmen. <p>Analysieren</p> <p>Die Studierenden können</p> <ul style="list-style-type: none"> ein geeignetes Modell für schlanke Bauteile anhand der Belastungsart und Geometrie auswählen. ein problemangepasstes Berechnungsverfahren zur Ermittlung von Reaktionsgrößen und Verformungen auch an statisch unbestimmten Systemen wählen. eine geeignete Festigkeitshypothese wählen. den relevanten Knickfall für gegebene Randbedingungen identifizieren. <p>Evaluiere (Beurteilen)</p> <p>Die Studierenden können</p> <ul style="list-style-type: none"> den Spannungszustand in einem Bauteil hinsichtlich Aspekten der Festigkeit bewerten. den Spannungszustand in einem schlanken Bauteil hinsichtlich Aspekten der Stabilität bewerten.
7	Voraussetzungen für die Teilnahme	<p>Organisatorisches:</p> <p>Alle Informationen zum Ablauf der Lehrveranstaltung werden über den StudOn-Kurs kommuniziert. Deshalb bitten wir Sie, sich unter https://www.studon.fau.de/cat5282.html einzuschreiben. Der Beitritt ist nicht, wie sonst üblich, passwortgeschützt, sondern erfolgt nach Bestätigung durch den Dozenten. Dies geschieht mitunter nicht umgehend, aber rechtzeitig vor dem ersten Termin. Wir bitten um Ihr Verständnis.</p>
8	Einpassung in Studienverlaufsplan	Semester: 4

9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	in jedem Semester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 105 h Eigenstudium: 120 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • Gross, Hauger, Schnell, Wall: Technische Mechanik 1, Berlin:Springer 2006 • Gross, Hauger, Schnell, Wall: Technische Mechanik 2, Berlin:Springer 2007

1	Modulbezeichnung 95010	Grundlagen der Elektrischen Antriebstechnik Foundations of drive engineering	5 ECTS
2	Lehrveranstaltungen	Praktikum: Praktikum Grundlagen der Elektrischen Antriebstechnik (3 SWS, SoSe 2025)	1,5 ECTS
3	Lehrende	Veronika Solovieva Philipp Sisterhenn Shan Jiang Babak Dianati Zidan Zhao Alexander Pfannschmidt Prof. Dr.-Ing. Susanne Lehner	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Ingo Hahn	
5	Inhalt	<p>*Grundlagen der Elektrischen Antriebstechnik*</p> <p>*Einleitung; Grundlagen:* Leistung und Wirkungsgrad, Physikalische Grundgesetze, Induktivitäten</p> <p>*Gleichstromantriebe:* Gleichstrommotor, Konventionelle Drehzahlstellung</p> <p>*Drehstromantriebe:* Grundlagen und Drehfeld, Synchronmaschine, Asynchronmaschine, Konventionelle Drehzahlstellung</p> <p>*Praktikum Grundlagen der Elektrischen Antriebstechnik*</p> <p>Die Studierenden führen im Labor drei Versuche durch: V1 Gleichstromantrieb V2 Asynchronmaschine am Pulsumrichter V3 Asynchronmaschine - Stationäres Betriebsverhalten</p> <p>Vor dem jeweiligen Versuch bereiten die Teilnehmer sich anhand der Unterlagen des Moduls "Grundlagen der elektrischen Antriebstechnik" und spezieller Unterlagen zum Versuch vor. Nach dem Versuch ist eine Ausarbeitung anzufertigen.</p>	
6	Lernziele und Kompetenzen	<p>*Grundlagen der Elektrischen Antriebstechnik*</p> <p>Kenntnisse und Verständnis der grundsätzlichen Funktionsweise elektrischer Maschinen, deren stationären Betrieb und die konventionelle (verlustbehaftete) Drehzahlstellung</p> <p>*Die Studierenden*</p> <ul style="list-style-type: none"> • haben einen Überblick über die Elektrische Antriebstechnik • kennen die Einsatzgebiete und Verbreitung elektrischer Antriebe sowie deren wirtschaftliche und gesellschaftliche Bedeutung • kennen die am weitesten verbreiteten Typen elektrischer Maschinen • kennen und verstehen bei Gleichstrommaschinen <p>- Aufbau und Funktionsweise - fachspezifische Begriffe - Feldverläufe in der Maschine - Kommutierung - beschreibende Spannungs- und Drehmomentgleichungen - stationären Betrieb und Betriebskennlinien - Drehmoment- Drehzahlkennlinie</p>	

		<ul style="list-style-type: none"> ◦ kennen passive und aktive Halbleiterbauelemente - Diode - Thyristor - Bipolartransistor - IGBT - MOS-Transistor - GTO-Thyristor kennen und verstehen bei Gleichstromantrieben - Aufbau und Funktionsweise - Methoden zur Drehzahlstellung und Spannungsstellung - elektronische Schaltungen und Ansteuermethoden - Gleichrichter - Tiefsetzsteller - Methode der Pulsweitenmodulation kennen und verstehen bei Drehstromantrieben - Aufbau und Funktionsweise - Methoden zur Drehzahlstellung und Spannungsstellung - elektronische Schaltungen und Ansteuermethoden - 3-phasiger Gleichrichter - ungesteuert/gesteuert - 3-phasiger Wechselrichter - Aufbau und Funktionsweise - Pulsweitenmodulation - Sinus-Dreieck-Modulation - U/f-Betrieb wenden die theoretische Grundlagen an und berechnen einfache lineare und nichtlineare Magnetkreise berechnen induzierte Spannungen und Drehmomente bei elektrischen Maschinen ermitteln auf Basis gegebener Kennwerte Arbeitspunkte auf der Betriebskennlinie erklären die Funktionsweise des Tiefsetzstellers und des gesteuerten 3-phasigen Gleichrichters entwickeln und wenden Zeigerdiagramme zur Darstellung des stationären Betriebsverhaltens von Drehfeldmaschinen an *Praktikum Grundlagen der Elektrischen Antriebstechnik* Die Grundkenntnisse aus Vorlesung und Übung "Grundlagen der Elektrischen Antriebstechnik" sollen gefestigt und erweitert werden. Der praktische Umgang mit elektrischen Antrieben und der zugehörigen Messtechnik soll erlernt werden.
7	Voraussetzungen für die Teilnahme	<p>*Vorlesung Grundlagen der Elektrischen Antriebstechnik*</p> <p>Die bestandene Prüfung im Fach "Grundlagen der Elektrotechnik I und II" ist ausdrücklich empfohlene Voraussetzung für die Teilnahme am Praktikum "Grundlagen der Elektrischen Antriebstechnik". Siehe Hinweis auf den Anschlagbrettern des Lehrstuhls und auf der Homepage.</p> <p>*Praktikum Grundlagen der Elektrischen Antriebstechnik*</p> <p>*Zulassungsbeschränkung:* Teilnahme ist *auch ohne* bestandener bzw. abgelegter Prüfung im Fach "Grundlagen der Elektrischen Antriebstechnik" möglich.</p> <p>Grundlagen der Elektrotechnik I und II</p> <p>*Anmeldung über StudOn*</p> <p>http://www.studon.uni-erlangen.de/crs687913.html</p> <p>*Ansprechpartner: Shima Khoshzaman, M.Sc.*</p>
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010

10	Studien- und Prüfungsleistungen	Praktikumsleistung Klausur mit MultipleChoice (90 Minuten) Das Praktikum besteht aus: <ul style="list-style-type: none"> • häusliche Vorbereitung • Dokumentation • 3 Versuche
11	Berechnung der Modulnote	Praktikumsleistung (bestanden/nicht bestanden) Klausur mit MultipleChoice (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	2 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 92670	Sensorik Sensor technology	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Philipp Beckerle	
5	Inhalt	<ul style="list-style-type: none"> • Einführung in die Sensorik • Wandlerprinzipien • Sensor-Parameter • Sensor-Technologien • Messung mechanischer Größen • Chemo- und Biosensoren 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • geben die Grundbegriffe und -strukturen der Sensorik und Aktorik wieder • klassifizieren Sensoren anhand unterschiedlicher Gesichtspunkte • beschreiben, skizzieren und vergleichen die behandelten Wandlerprinzipien und Technologien zur Herstellung von Sensoren • kennen die behandelten Sensor-Parameter und beurteilen Sensoren anhand dieser • beschreiben und charakterisieren die behandelten Sensoren zur Messung mechanischer Größen • analysieren Elemente der Sensor- und Aktortechnik sowie Schaltungen zur Weiterverarbeitung und Auswertung von Messgrößen • zeigen mögliche Fehlerquellen der Sensorik auf und arbeiten Strategien zur Minimierung der Fehler aus 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	NF Mechatronics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Wintersemester	

13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Tränkler, Hans-Rolf: "Sensortechnik - Handbuch für Praxis und Wissenschaft", 2. Aufl. 2014, Springer Vieweg Hering, Eckert: "Sensoren in Wissenschaft und Technik - Funktionsweise und Einsatzgebiete", 2. Aufl. 2018, Springer Fachmedien Wiesbaden Mitchell, H. B.: "Data fusion: concepts and ideas", 2012, Springer

1	Modulbezeichnung 97190	Technische Schwingungslehre Mechanical vibrations	5 ECTS
2	Lehrveranstaltungen	Tutorium: Tutorium zur Technischen Schwingungslehre (2 SWS) Vorlesung: Technische Schwingungslehre (2 SWS) Übung: Übungen zur Technischen Schwingungslehre (2 SWS)	- - -
3	Lehrende	Özge Akar Prof. Dr.-Ing. Kai Willner	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Kai Willner
5	Inhalt	<p>Charakterisierung von Schwingungen Mechanische und mathematische Grundlagen</p> <ul style="list-style-type: none"> • Bewegungsgleichungen • Darstellung im Zustandsraum <p>Allgemeine Lösung zeitinvarianter Systeme</p> <ul style="list-style-type: none"> • Anfangswertproblem • Fundamentalmatrix • Eigenwertaufgabe <p>Freie Schwingungen</p> <ul style="list-style-type: none"> • Eigenwerte und Wurzelortskurven • Zeitverhalten und Phasenportraits • Stabilität <p>Erzwungene Schwingungen</p> <ul style="list-style-type: none"> • Sprung- und Impulserregung • harmonische und periodische Erregung • Resonanz und Tilgung <p>Parametererregte Schwingungen</p> <ul style="list-style-type: none"> • Periodisch zeitinvariante Systeme <p>Experimentelle Modalanalyse</p> <ul style="list-style-type: none"> • Bestimmung der Übertragungsfunktionen • Bestimmung der modalen Parameter • Bestimmung der Eigenmoden
6	Lernziele und Kompetenzen	<p>Fachkompetenz Wissen</p> <ul style="list-style-type: none"> • Die Studierenden kennen verschiedene Methoden die Bewegungsdifferentialgleichungen diskreter Systeme aufzustellen. • Die Studierenden kennen verschiedene Schwingungsarten und Schwingertypen. • Die Studierenden kennen die Lösung für die freie Schwingung eines linearen Systems mit einem Freiheitsgrad und die entsprechenden charakteristischen Größen wie Eigenfrequenz und Dämpfungsmaß. • Die Studierenden kennen eine Reihe von analytischen Lösungen des linearen Schwingers mit einem Freiheitsgrad für spezielle Anregungen.

- Die Studierenden kennen die Darstellung eines Systems in physikalischer Darstellung und in Zustandsform.
- Die Studierenden kennen die Darstellung der allgemeinen Lösung eines linearen Systems mit mehreren Freiheitsgraden in Zustandsform.
- Die Studierenden kennen das Verfahren der modalen Reduktion.
- Die Studierenden kennen Verfahren zur numerischen Zeitschrittintegration bei beliebiger Anregung.
- Die Studierenden kennen die Definition der Stabilität für lineare Systeme.

Verstehen

- Die Studierenden können ein gegebenes diskretes Schwingungssystem anhand des zugrundeliegenden Differentialgleichungssystems einordnen und klassifizieren.
- Die Studierenden verstehen den Zusammenhang zwischen der physikalischen Darstellung und der Zustandsdarstellung und können die Vor- und Nachteile der beiden Darstellungen beschreiben.
- Die Studierenden verstehen die Bedeutung der Fundamentalmatrix und können diese physikalisch interpretieren.
- Die Studierenden verstehen die Idee der modalen Reduktion und können ihre Bedeutung bei der Lösung von Systemen mit mehreren Freiheitsgraden erläutern.
- Die Studierenden können den Stabilitätsbegriff für lineare Systeme erläutern.

Anwenden

- Die Studierenden können die Bewegungsdifferentialgleichungen eines diskreten Schwingungssystem auf verschiedenen Wegen aufstellen
- Die Studierenden können die entsprechende Zustandsdarstellung aufstellen.
- Die Studierenden können fuer einfache lineare Systeme die Eigenwerte und Eigenvektoren von Hand ermitteln und kennen numerische Verfahren zur Ermittlung der Eigenwerte und -vektoren bei großen Systemen.
- Die Studierenden können aus den Eigenwerten und -vektoren die Fundamentalmatrix bestimmen und für gegebene Anfangsbedingungen die Lösung des freien Systems bestimmen.
- Die Studierenden können ein lineares System mit mehreren Freiheitsgraden modal reduzieren.
- Die Studierenden können die analytische Loesung eines System mit einem Freiheitsgrad für eine geeignete Anregung von Hand bestimmen und damit die Lösung im Zeitbereich und in der Phasendarstellung darstellen.

Analysieren

		<ul style="list-style-type: none"> Die Studierenden können problemgerecht zwischen physikalischer Darstellung und Zustandsdarstellung wählen und die entsprechenden Verfahren zur Bestimmung der Eigenlösung und gegebenenfalls der partikulären Lösung einsetzen. <p>Evaluieren (Beurteilen)</p> <ul style="list-style-type: none"> Die Studierenden können anhand der Eigenwerte bzw. der Wurzelorte das prinzipielle Lösungsverhalten eines linearen Schwingungssystems beurteilen und Aussagen über die Stabilität eines Systems treffen.
7	Voraussetzungen für die Teilnahme	<p>Empfohlen: Kenntnisse aus dem Modul "Dynamik starrer Körper"</p> <p>Organisatorisches: Alle Informationen zum Ablauf der Lehrveranstaltung werden über den StudOn-Kurs kommuniziert. Deshalb bitten wir Sie, sich unter https://www.studon.fau.de/cat5282.html einzuschreiben. Der Beitritt ist nicht, wie sonst üblich, passwortgeschützt, sondern erfolgt nach Bestätigung durch den Dozenten. Dies geschieht mitunter nicht umgehend, aber rechtzeitig vor dem ersten Termin. Wir bitten um Ihr Verständnis. We will communicate all information about the lecture schedule via the StudOn course. Therefore, we ask you to enroll at https://www.studon.fau.de/cat5282.html. The entry is not password-protected, as usual, but takes place after confirmation by the lecturer. The acceptance may not happen immediately, but in time for the first class. We ask for your understanding.</p>
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>Klausur (90 Minuten)</p> <p>Technische Schwingungslehre (Prüfungsnummer: 71901) Prüfungsleistung, Klausur, Dauer (in Minuten): 90, benotet</p>
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 60 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Magnus, Popp: Schwingungen, Stuttgart:Teubner 2005

1	Modulbezeichnung 97270	Mehrkörperdynamik Multibody dynamics	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Sigrid Leyendecker	
5	Inhalt	<ul style="list-style-type: none"> • Kinematik für Systeme gekoppelter starrer Körper • Dreidimensionale Rotationen • Newton-Euler-Gleichungen des starren Körpers • Bewegungsgleichungen für Systeme gekoppelter Punktmassen/starrer Körper • Parametrisierung in generalisierten Koordinaten und in redundanten Koordinaten • Untermannigfaltigkeiten, Tangential- und Normalraum • Nichtinertialkräfte • Holonome und nicht-holonome Bindungen • Bestimmung der Reaktionsgrößen in Gelenken • Indexproblematik bei numerischen Lösungsverfahren für nichtlineare Bewegungsgleichungen mit Bindungen • Steuerung in Gelenken • Topologie von Mehrkörpersystemen 	
6	Lernziele und Kompetenzen	<p>Fachkompetenz Wissen Die Studierenden:</p> <ul style="list-style-type: none"> • kennen das innere, äußere und dyadische Produkt von Vektoren. • kennen die einfache und zweifache Kontraktion von Tensoren. • kennen den Satz von Euler für die Fixpunktdrehung. • kennen mehrere Möglichkeiten, dreidimensionale Rotationen zu parametrisieren (etwa Euler-Winkel, Cardan-Winkel oder Euler-Rodrigues-Parameter). • kennen die Problematik mit Singularitäten bei Verwendung dreier Parameter. • kennen die $SO(3)$ und $so(3)$. • kennen den Zusammenhang zwischen Matrixexponentialfunktion und Drehzeiger. • kennen die Begriffe Untermannigfaltigkeit, Tangential- und Normalraum. • kennen die Begriffe Impuls und Drall eines starren Körpers. • kennen den Aufbau der darstellenden Matrix des Trägheitstensors eines starren Körpers. • kennen den Satz von Huygens-Steiner. • kennen die Begriffe holonom-skleronome und holonom-rheonome Bindungen. 	

- kennen den Begriff des differentiellen Indexes eines differential-algebraischen Gleichungssystems.
- kennen die expliziten und impliziten Reaktionsbedingungen in den Gelenken von Mehrkörpersystemen.
- kennen aus Dreh- und Schubgelenken zusammensetzbare Gelenke.
- kennen niedrige und höhere Elementenpaare.
- kennen den Unterschied zwischen offenen und geschlossenen Mehrkörpersystemen.
- kennen den Satz über Hauptachsentransformation symmetrischer reeller Matrizen.
- kennen die nichtlinearen Effekte bei der Kreiselbewegung.

Verstehen

Die Studierenden:

- verstehen den Unterschied zwischen (physikalischen) Tensoren/Vektoren und (mathematischen) Matrizen/Tripeln.
- verstehen den Relativkinematik-Kalkül auf Lage, Geschwindigkeits- und Beschleunigungsebene.
- verstehen, wie sich die Matrix des Trägheitstensors bei Translation und Rotation transformiert.
- verstehen die Trägheitseigenschaften eines starren Körpers.
- verstehen den Unterschied zwischen eingprägten Kräften und Reaktionskräften.
- verstehen den Unterschied zwischen expliziten und impliziten Reaktionsbedingungen.
- verstehen den Impuls- und Drallsatz (Newton-Euler-Gleichungen) für den starren Körper.
- verstehen die mechanischen Effekte, die auftretende Nichtinertialkräfte bewirken.
- verstehen, dass die $SO(3)$ (multiplikative) Gruppenstruktur, die $so(3)$ (additive) Vektorraumstruktur trägt.
- verstehen, warum dreidimensionale Rotationen nicht kommutativ sind.
- verstehen, welche Drehungen um Hauptachsen stabil, welche instabil sind.
- verstehen das Verfahren der Indexreduktion für die auftretenden differential-algebraischen Systeme.
- verstehen das Phänomen des Wegdriftens bei indexreduzierten Formulierungen der Bewegungsgleichungen.
- verstehen, wie man dem Wegdriften entgegenwirken kann.
- verstehen die analytische Lösung der Euler-Gleichungen des kräftefreien symmetrischen Kreisels.
- verstehen die Poincaré-Beschreibung des kräftefreien Kreisels.
- verstehen die Beweise der zugehörigen analytischen Zusammenhänge, einschließlich der Voraussetzungen.

Anwenden

Die Studierenden:

- können Koeffizienten von Vektoren und Tensoren zwischen verschiedenen Koordinatensystemen transformieren.

- können den Relativkinematik-Kalkül anwenden, d.h. mehrere Starrkörperbewegungen miteinander verketten.
- können Rotationen aktiv und passiv interpretieren.
- können allgemein mit generalisierten Koordinaten umgehen.
- können die Winkelgeschwindigkeit zu einer gegebenen Parametrisierung der Rotationsmatrix berechnen.
- können zu einer gegebenen Untermannigfaltigkeit Normal- und Tangentialraum bestimmen.
- können den Impuls- und Drallsatz auf starre Körper anwenden.
- können die Bindungen auf Lage-, Geschwindigkeits und Beschleunigungsebene bestimmen.
- können die Bewegungsgleichungen dynamischer Systeme in minimalen generalisierten Koordinaten aufstellen.
- können die Bewegungsgleichungen dynamischer Systeme in redundanten Koordinaten aufstellen.
- können letztere in erstere überführen.
- können die Lagrange-Multiplikatoren sowie die zugehörigen Reaktionskräfte systematisch als Funktion der Lage- und Geschwindigkeitsgrößen berechnen.
- können geeignete Nullraum-Matrizen finden.
- können die Reaktionskräfte in den Bewegungsgleichungen via Nullraummatrix eliminieren.
- können das Verfahren der Indexreduktion auf die Bewegungsgleichungen in redundanten Koordinaten anwenden.
- können den Index alternativer Formulierungen der Bewegungsgleichungen (etwa GGL-Formulierung) berechnen.
- können das Phänomen des Wegdriftens durch Projektionsverfahren oder Baumgarte-Stabilisierung unterbinden.
- können die translatorische und rotatorische Energie eines starren Körpers berechnen.
- können Hauptträgheitsmomente und -richtungen via Hauptachsentransformation ermitteln.
- können Trägheitsmomente einfacher Körper durch Volumenintegration berechnen.
- können den Satz von Huygens-Steiner anwenden.
- können den Freiheitsgrad holonomer Systeme bestimmen.
- können skleronome und rheonome Gelenke modellieren.
- können Mehrkörpermodelle topologisch und kinematisch klassifizieren.
- können analytische Lösungen der Bewegungsgleichungen (etwa Foucault-Pendel, symmetrischer Kreisel) durch Differentiation verifizieren.
- können die dynamische rechte Seite der Bewegungsgleichungen in Matlab implementieren und mit Standard-Zeitintegrationsverfahren lösen.
- können die Beweise der wichtigsten mathematischen Sätze eigenständig führen.

		<p>Analysieren</p> <p>Die Studierenden:</p> <ul style="list-style-type: none"> • können analytische Lösungen der Bewegungsgleichungen (etwa Foucault-Pendel, symmetrischer Kreisel) eigenständig durch Integration bestimmen. • können die Auswirkungen der Zentrifugalmomente eines starren Körpers bei der Auslegung von Maschinen qualitativ und quantitativ beurteilen. <p>Erschaffen</p> <p>Die Studierenden:</p> <ul style="list-style-type: none"> • können Mehrkörpermodelle realer Maschinen mit starren Körpern, Kraftelementen und Gelenken selbstständig aufbauen. • können deren Dynamik durch numerische Simulation analysieren.
7	Voraussetzungen für die Teilnahme	Empfohlen: Modul Dynamik starrer Körper
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (120 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • Schiehlen, Eberhard: Technische Dynamik. Teubner, 2004 • Woernle: Mehrkörpersysteme. Eine Einführung in die Kinematik und Dynamik von Systemen starrer Körper. Springer, 2011

1	Modulbezeichnung 97330	Strömungsmechanik II Fluid mechanics II	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Andreas Wierschem	
5	Inhalt	<ul style="list-style-type: none"> • Dimensionsanalyse und Ähnlichkeitstheorie • schleichende Strömungen • zeitabhängige Strömungen • Potentialströmungen • Grenzschichtströmungen • Turbulenz • kompressible Strömungen <p>Übungen ergänzen die Vorlesung. Studierende werden angeleitet, das erhaltene Wissen anzuwenden, strömungsmechanische Problemstellungen zu bewerten und Lösungen zu entwickeln.</p>	
6	Lernziele und Kompetenzen	<p>Aufbauend auf Kenntnissen reibungsbehafteter Strömungen bietet die Vorlesung eine systematische Vertiefung in wesentliche Bereiche der Strömungsmechanik.</p> <p>Die Studierenden:</p> <ul style="list-style-type: none"> • verfügen über einen Überblick über wesentliche Bereiche der Strömungsmechanik und verstehen ihre Bedeutung und Anwendung in der Strömungsmechanik • können die Bedeutung der unterschiedlichen Strömungsbereiche sowohl in der natürlichen Umgebung als auch in ingenieurwissenschaftlichen Problemstellungen nachvollziehen • sind fähig, strömungsmechanische Problemstellungen zu bewerten und Lösungen zu entwickeln • können die erworbenen Fachkenntnisse mit geeigneten Methoden und Vorgehensweisen an Hand von Beispielen praktisch anwenden. 	
7	Voraussetzungen für die Teilnahme	Empfohlen: Strömungsmechanik (CBI, CEN) oder Strömungsmechanik I für Maschinenbau und Energietechnik.	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	schriftlich oder mündlich mündlich, 30 min	
11	Berechnung der Modulnote	schriftlich oder mündlich (100%) mündlich, 100 %	

12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • J. H. Spurk, N. Aksel: Strömungslehre: Einführung in die Theorie der Strömungen, 8. Auflage, Springer-Verlag Berlin, Heidelberg, New York, 2010 • F. Durst, Grundlagen der Strömungsmechanik - Eine Einführung in die Theorie der Strömungen in Fluiden, Springer, 2006 • P. K. Kundu, Fluid Mechanics, 5th Ed., Academic Press, 2012 • F. M. White, Fluid Mechanics, 7th Rev. Ed., McGraw Hill, 2011

1	Modulbezeichnung 97360	Digitale Regelung Digital control	5 ECTS
2	Lehrveranstaltungen	Vorlesung mit Übung: Digitale Regelung (4 SWS)	5 ECTS
3	Lehrende	Dr.-Ing. Andreas Michalka	

4	Modulverantwortliche/r	Dr.-Ing. Andreas Michalka	
5	Inhalt	<p>Es werden Aufbau u. mathematische Beschreibung digitaler Regelkreise für LZI-Systeme sowie Verfahren zu deren Analyse und Synthese betrachtet:</p> <ul style="list-style-type: none"> • quasikontinuierliche Beschreibung und Regelung der Strecke unter Berücksichtigung der DA- bzw. AD-Umsetzer • zeitdiskrete Beschreibung der Regelstrecke als Zustandsgleichung oder z-Übertragungsfunktion • Analyse von Abtastsystemen, Stabilität, Steuer- und Beobachtbarkeit • Regelungssynthese: Steuerungsentwurf, Zustandsregelung und Beobachterentwurf, Störungen im Regelkreis, Berücksichtigung von Totzeiten, Intersampling-Verhalten". 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • erläutern Aufbau und Bedeutung digitaler Regelkreise. • leiten mathematische Beschreibungen des Abtastsystems in Form von Zustandsgleichungen oder z-Übertragungsfunktionen her. • analysieren Abtastsysteme und konzipieren digitale Regelungssysteme auf Basis quasikontinuierlicher sowie zeitdiskreter Vorgehensweisen. • entwerfen Steuerungen, Regelungen und Beobachter und bewerten die erzielten Ergebnisse. • diskutieren abtastregelungsspezifische Effekte und bewerten Ergebnisse im Vergleich mit dem kontinuierlichen Systemverhalten. 	
7	Voraussetzungen für die Teilnahme	<p>Es wird empfohlen folgende Module zu absolvieren, bevor dieses Modul belegt wird:</p> <ul style="list-style-type: none"> • Regelungstechnik A (Grundlagen) (RT A) oder Einführung in die Regelungstechnik (ERT) • Regelungstechnik B (Zustandsraummethoden) (RT B) 	
8	Einpassung in Studienverlaufsplan	Semester: 6	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Schriftliche Prüfung (Klausur, mit 90 Minuten Dauer).	
11	Berechnung der Modulnote	Klausur (100%)	
12	Turnus des Angebots	nur im Sommersemester	

13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 275245	Heterogene Rechnerarchitekturen Online Heterogeneous computing architectures online	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Heterogene Rechnerarchitekturen Online (0 SWS)	-
3	Lehrende	Johannes Kliemt Prof. Dr.-Ing. Dietmar Fey	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Dietmar Fey	
5	Inhalt	<p>Whereas heterogeneous architectures and parallel computing has filled an academic niche in the past it has become now a commodity technique with the rising of multi-core processors and programmable graphic cards. Even FPGAs play a role hereby in a certain extent due to their increasing importance as accelerator hardware what is clearly observable in the scientific community. However, on one side parallel hardware like multi-core and GPUs are now available nearly for everybody and not only for a selected selection of people, who have access to a parallel supercomputer. On the other side the knowledge about programming of this commodity hardware, and we mean here in particular hardware-orientated programming in order to squeeze out all offered GFlops and TFlops of such hardware, is still missing as well as the knowledge about the architecture details.</p> <p>To overcome this lack we offer this course HETRON. The e-learning course HETRON for the exploitation of parallel and heterogeneous computer architectures) focuses on two main topics which are closely related to each other. This concerns on one side the benefits of using different kinds of multi-core processors and parallel architectures built-up on base of these multicore processors. These architectures differ among each other in the number and in the complexity of its single processing nodes. We distinguish between systems consisting of a large number of simpler, so called fine-grained, processor cores vs. systems consisting of a smaller number of more complex, so called coarse-grained, processor cores.</p> <p>On the other side we lay our focus on that we want to do with these different heterogeneous parallel architectures, namely the execution of parallel programs. Of course this requires the use of parallel programming languages and environments, like CUDA or OpenMP. However, besides these questions of using the right syntax and the right compiler switches to optimize a parallel program it is a pre-requisite to understand how</p>	

		<p>parallel computing really works. This refers (i) to the comprehension which basic mechanisms of parallel computing exist, (ii) where are the limits of getting more performance with parallel computing and (iii) in what context stand these mechanisms to heterogeneous architectures. In other words it handles the question which architecture is the best one for a certain parallelization technique. To teach these three topics, is one main goal we pursuit with the course HETRON, and of course, this more fundamental basics of heterogeneous and parallel computing have to be proven by means of concrete application examples to deepen the acquired knowledge about heterogeneous architectures and parallel computing principles.</p>
6	Lernziele und Kompetenzen	<p>Die Studierenden ...</p> <ul style="list-style-type: none"> ...verstehen die Notwendigkeit sowie grundlegende Anwendungsfälle für heterogene Rechnerarchitekturen. ...können den grundlegenden Aufbau und das Zusammenspiel der Komponenten heterogener Rechnerarchitekturen erklären. ...erläutern grundsätzliche Parallelisierungsprinzipien wie Amdahls Law, High-Performance- und High-Throughput-Computing sowie Parallelisierungsstrategien. ...können einfache Programme mit Hilfe der vermittelten Parallelisierungsprinzipien (Amdahls Law, High-Performance- und High-Throughput-Computing) analysieren und entsprechende Parallelisierungsstrategien entwickeln. ...erklären den Aufbau sowie Stärken und Schwächen von verschiedenen Architekturen wie CPUs, GPUs, Many-Core Prozessoren und FPGAs. ...implementieren ausgewählte Anwendungsbeispiele (SHA256 Algorithmus, Ising-Modell und Fast-Fourier-Transformation) auf oben genannte Architekturen. ...erforschen und bewerten verschiedener Parallelsierungstechniken in Abhängigkeit der Anwendung und der Architektur. ...erläutern die Grundlagen des Grid- und Cloud-Computings ...sind in der Lage parallele Berechnungen (SHA256) im Grid umzusetzen.
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Variabel (60 Minuten)
11	Berechnung der Modulnote	Variabel (100%)

12	Turnus des Angebots	in jedem Semester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 0 h Eigenstudium: 150 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch oder Englisch
16	Literaturhinweise	

1	Modulbezeichnung 299892	Informationsvisualisierung Information visualization	5 ECTS
2	Lehrveranstaltungen	Übung: Übung zur Informationsvisualisierung (2 SWS) Vorlesung: Informationsvisualisierung (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Dr. Roberto Grosso	

4	Modulverantwortliche/r	Dr. Roberto Grosso	
5	Inhalt	<p>Aufgrund der rasanten Entwicklung der Informationstechnologie sind wir mit einer noch nie dagewesenen Flut an Daten konfrontiert. Informationsvisualisierung befasst sich mit der graphischen Darstellung abstrakter Daten, die keine räumliche Struktur aufweisen. Die Visualisierung abstrakter Daten nutzt visuelle Metaphern und Interaktion, um Information aus den Daten zu extrahieren. Typische Anwendungsszenarien sind die Analyse von Finanztransaktionen oder sozialen Netzwerken, Geographie, Textanalyse oder Visualisierung von Software-Quellcode.</p> <p>In dieser Vorlesung werden unterschiedliche Techniken vorgestellt, um verschieden Arten von Daten zu visualisieren.</p> <p>Insbesondere werden folgende Themen behandelt:</p> <ul style="list-style-type: none"> • Graphen und Netzwerke • Dynamische Graphen • Hierarchien und Bäume • Multivariate Daten • Time-Series Daten • Textvisualisierung 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <p>Wissen</p> <ul style="list-style-type: none"> • zählen Datentypen der Informationsvisualisierung auf • nennen Techniken zur Visualisierung unterschiedlicher Datentypen der Informationsvisualisierung • beschreiben Anwendungsfällen für die unterschiedlichen Datentypen der Informationsvisualisierung <p>Verstehen</p> <ul style="list-style-type: none"> • stellen Algorithmen der Informationsvisualisierung dar und erläutern ihre Eigenschaften, Vorteile und Nachteile • illustrieren Techniken zu Auswertung und Analyse von Daten der Informationsvisualisierung • implementieren die vorgestellten Algorithmen in JavaScript <p>Anwenden</p> <ul style="list-style-type: none"> • wenden Algorithmen zur Visualisierung unterschiedlichen Daten an • erklären und charakterisieren Techniken der Informationsvisualisierung • <p>Analysieren</p> <ul style="list-style-type: none"> • klassifizieren Algorithmen zur Visualisierung multivariater Daten, Netzwerke, Hierarchien und Text und erklären ihrer Funktionsweise 	

		<ul style="list-style-type: none"> erkunden die Effizienz der vorgestellten Algorithmen für unterschiedlichen Anwendungsmöglichkeiten <p>Evaluieren</p> <ul style="list-style-type: none"> bewerten Anwendbarkeit und Performance spezieller Algorithmen der Informationsvisualisierung vergleichen Methoden zur Analyse und Auswertung von Daten der Informationsvisualisierung überprüfen die Anwendbarkeit der diskutierten Techniken für unterschiedliche, speziell ausgewählten Fälle
7	Voraussetzungen für die Teilnahme	Die Programmieraufgaben werden in JavaScript implementiert.
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	elektronische Prüfung mit MultipleChoice (90 Minuten)
11	Berechnung der Modulnote	elektronische Prüfung mit MultipleChoice (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	<p>Information Visualization</p> <ul style="list-style-type: none"> Robert Spence: Information Visualization: Design for Interaction Stuart K. Card, Jock Mackinlay, Ben Shneiderman: Readings in Information Visualization – Using Vision to Think Benjamin B. Bederson, Ben Shneiderman: The Craft of Information Visualization – Readings and Reflections Tamara Munzner: Visualization Analysis and Design Colin Ware: Information Visualization, Perception for Design (third edition) Ricardo Mazza: Introduction to Information Visualization Robert Spence: Information Visualization - An Introduction <p>Networks / Graphs</p> <ul style="list-style-type: none"> <ul style="list-style-type: none"> Graph Theory, Reinhard Diestel Graphentheorie, Peter Tittmann Graphs, Networks and Algorithms, Dieter Jungnickel <ul style="list-style-type: none"> Networks, 2nd Edition, Mark Newman

1	Modulbezeichnung 333815	Rechnerarchitektur (Vorlesung mit Übung und Rechnerübung) Computer architecture	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Dietmar Fey
5	Inhalt	<p>Die Vorlesung baut auf die in den Grundlagen der Rechnerarchitektur und -organisation vermittelten Inhalte auf und setzt diese mit weiterführenden Themen fort. Es werden zunächst grundlegende fortgeschrittene Techniken bei Pipelineverarbeitung und Cachezugriffen in modernen Prozessoren und Parallelrechnern behandelt. Ferner wird die Architektur von Spezialprozessoren, z.B. DSPs und Embedded Prozessoren behandelt. Es wird aufgezeigt, wie diese Techniken in konkreten Architekturen (Intel Nehalem, GPGPU, Cell BE, TMS320 DSP, Embedded Prozessor ZPU) verwendet werden. Zur Vorlesung werden eine Tafel- und eine Rechnerübung angeboten. Die Rechnerübung erfordert 11 erfolgreich abgeschlossene Übungsaufgaben, diese gehen mit 10% in die Modulgesamtnote ein. Die verbleibenden 90% werden durch die mündliche Prüfung bestimmt. Insgesamt werden 7,5 ECTS erworben. In den Tafelübungen werden die in der Vorlesung vermittelten Techniken durch zu lösende Aufgaben vertieft. In der Rechnerübung soll u.a. ein einfacher Vielkern-Prozessor auf Basis des ZPU-Prozessors mit Simulationswerkzeugen aufgebaut werden. Im Einzelnen werden folgende Themen behandelt:</p> <ul style="list-style-type: none"> • Organisationsaspekte von CISC und RISC-Prozessoren • Behandlung von Hazards in Pipelines • Fortgeschrittene Techniken der dynamischen Sprungvorhersage • Fortgeschrittenen Cachetechniken, Cache-Kohärenz • Ausnutzen von Cacheeffekten • Architekturen von Digitalen Signalprozessoren • Architekturen homogener und heterogener Multikern-Prozessoren (Intel Corei7, Nvidia GPUs, RISC-V) • Architektur von Parallelrechnern (Clusterrechner, Superrechner) • Effiziente Hardware-nahe Programmierung von Multikern-Prozessoren (OpenMP, SSE, CUDA) • Leistungsmodellierung und -analyse von Multikern-Prozessoren (Roofline-Modell)
6	Lernziele und Kompetenzen	<p>Fachkompetenz Wissen Lernende können Wissen abrufen und wiedergeben. Sie können konkrete Einzelheiten wie Begriffe, Definitionen, Fakten, und Abläufe in einem Prozessor darlegen.</p>

		<p>Verstehen Lernende können Beispiele für Rechnerarchitekturen anführen, sie sind in der Lage, Schaubilder von Prozessoren zu interpretieren und die Abläufe in eigenen Worten zu beschreiben.</p> <p>Anwenden Lernende können beim Erstellen eigener Programme durch Transfer des Wissens über Interna von Prozessorarchitekturen Optimierungen hinsichtlich des Laufzeitverhaltens vornehmen.</p> <p>Analysieren Lernende können zwischen verschiedenen Varianten von Lösungen einer Prozessorarchitektur klassifizieren, die Gründe für durchgeführte Entwurfsentscheidungen erschließen, Unterscheide gegenüberstellen und gegeneinander bewerten.</p> <p>Lern- bzw. Methodenkompetenz Lernende erwerben die Fähigkeit selbstständig Testprogramme zum Bewerten der Leistungsfähigkeit eines Prozessors zu erstellen.</p>
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Variabel
11	Berechnung der Modulnote	Variabel (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • Patterson/Hennessy: Computer Organization und Design • Hennessy/Patterson: Computer Architecture - A Quantitative Approach • Stallings: Computer Organization and Architecture • Martin: Rechnerarchitekturen

1	Modulbezeichnung 399607	High End Simulation in Practice High end simulation in practice (lecture with exercise)	7,5 ECTS
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	Modulverantwortliche/r	
5	Inhalt	keine Inhaltsbeschreibung hinterlegt!
6	Lernziele und Kompetenzen	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 6
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Portfolio
11	Berechnung der Modulnote	Portfolio (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	Dauer des Moduls	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 465562	Advanced Programming Techniques Advanced programming techniques (lecture and exercises)	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	apl. Prof. Dr. Harald Köstler	
5	Inhalt	<p>Der Inhalt der Vorlesung besteht aus zahlreichen fortgeschrittenen C++-Themen, die ausgerichtet sind auf die richtige und effiziente Nutzung von C++ für eine professionelle Softwareentwicklung.</p> <p>The content of the lecture will consist of various topics of advanced C++ programming, aimed at teaching the proper and efficient usage of C++ for professional software development.</p> <p>These are basic language concepts, the newer standards (starting from C++11), object oriented programming in C++, static and dynamic polymorphism, template metaprogramming, and C++ idioms and design patterns.</p> <p>A good preparation for the lecture is the C++ primer book from S. Lippman et al. One should at least have several hundred hours of programming experience in C/C++ or any related object oriented programming language. Knowledge of basic concepts like pointers, references, inheritance and polymorphism is required.</p>	
6	Lernziele und Kompetenzen	<p>Fachkompetenz</p> <p>Wissen Lernende können die grundlegenden Sprachkonstrukte in den verschiedenen C++ Standards wiedergeben. Students know the basic language constructs from different C++ standards.</p> <p>Verstehen Lernende verstehen das C++ Objektmodell und können es mit anderen Programmiersprachen vergleichen. Students understand the C++ object model and are able to compare it to other programming languages.</p> <p>Anwenden Lernenden können Standardalgorithmen in einer objektorientierten Programmiersprache implementieren. Students can implement standard algorithms in an object oriented programming language.</p> <p>Analysieren Lernende können gängige Design Patterns klassifizieren und deren Anwendbarkeit für bestimmte Probleme diskutieren. Students are able to classify common design patterns and to discuss their usability for certain problems.</p> <p>Evaluieren (Beurteilen)</p>	

		<p>Lernende können entscheiden, welches Software Design passend für eine bestimmte Aufgabe ist. Sie können auch den Implementierungsaufwand dafür abschätzen.</p> <p>Students can decide, which software design fits for a certain task. They are also able to estimate the programming effort for it.</p> <p>Erschaffen</p> <p>Lernende entwickeln selbständig in einer Gruppe ein größeres Softwarepaket im Bereich Simulation und Optimierung.</p> <p>Students develop together in a group a larger software project in the area of simulation and optimization on their own.</p>
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Portfolio (60 Minuten)
11	Berechnung der Modulnote	Portfolio (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 165 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> • S. Lippman: C++ Primer, Addison-Wesley • S. Meyers: Effective C++ Third Edition, Addison-Wesley • H. Sutter: Exceptional C++, Addison-Wesley

1	Modulbezeichnung 508483	Praktikum Photonik/Lasertechnik 2 Laboratory course: Photonics/Laser technology 2	2,5 ECTS
2	Lehrveranstaltungen	Praktikum: Gruppe A (2 SWS) Praktikum: Gruppe B (2 SWS) Praktikum: Gruppe C (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Jasper Freitag	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Bernhard Schmauß	
5	Inhalt	<p>In kleinen Gruppen zu 2-3 Studierenden werden acht Versuche zu folgenden Themen der Lasertechnik und Photonik durchgeführt:</p> <ul style="list-style-type: none"> • Polarisation - Doppelbrechung - Jones-Matrizen • Zeitliche Kohärenz - Michelson-Interferometer Linienbreiten • Räumliche Kohärenz - Beugung Doppelspalt • Leistungs-Laserdiode - Kennlinie Wellenlängenabstimmung • Lichtwellenmesstechnik - Wavemeter - OSA • EDFA - Erbium-dotierter Faserverstärker - Faser-Laser • Nd:YAG-Laser - Kennlinien - Resonator - Stabilität • Dynamik im Laser - Q-Switch - Spiking - Sättigbarer Absorber <p>Anhand der Versuche wird gelernt, moderne und komplexe laserbasierte Systeme in der Praxis einzusetzen, als Voraussetzung für viele Anwendungen in Wissenschaft und Technik. Derartige Systeme werden eingesetzt z.B. für die Präzisionsmesstechnik, in der industriellen Materialbearbeitung, in der Bioanalytik, für die Medizintechnik, in Geräten der Unterhaltungselektronik oder in der optischen Nachrichtentechnik.</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • vertiefen ihre wissenschaftlichen Kenntnisse im Bereich der komplexer photonischer Systeme durch praktische Experimente. • können fortgeschrittene technische und wissenschaftliche Experimente im Bereich Photonik / Lasertechnik selbstständig und in kooperativen Gruppen planen, durchführen und reflektieren. • können Sachverhalte und Ergebnisse der im Inhalt beschriebenen Experimente bewerten und vergleichen. • sind in der Lage, eigenständig Ideen zur Lösung komplexer technisch-wissenschaftlicher Messaufgaben im Bereich der Photonik und Lasertechnik zu entwickeln. 	
7	Voraussetzungen für die Teilnahme	<ul style="list-style-type: none"> • Photonik 1 • Photonik 2 (kann vorlesungsbegleitend besucht werden) 	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Praktikumsleistung	

11	Berechnung der Modulnote	Praktikumsleistung (bestanden/nicht bestanden)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 45 h Eigenstudium: 30 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	Träger, F. (Ed.): Handbook of Lasers and Optics, Springer Verlag, Berlin 2007. Eichler/Eichler: Laser. Springer Verlag, Berlin 2006. Reider, G.A.: Photonik. Springer Verlag, Berlin 2003. Bergmann, Schäfer: Lehrbuch der Experimentalphysik, Bd.3: Optik. DeGruyter 1993.

1	Modulbezeichnung 532733	Künstliche Intelligenz II Artificial intelligence II	7,5 ECTS
2	Lehrveranstaltungen	Vorlesung: Artificial Intelligence II (4 SWS)	-
3	Lehrende	Prof. Dr. Michael Kohlhase	

4	Modulverantwortliche/r	Prof. Dr. Michael Kohlhase
5	Inhalt	Dieses Modul beschäftigt sich mit den Grundlagen der Künstlichen Intelligenz (KI), insbesondere mit Techniken des Schließens unter Unsicherheit, des maschinellen Lernens und der Sprachverarbeitung. Das Modul baut auf dem Modul Künstliche Intelligenz I vom Wintersemester auf und führt dieses weiter.
6	Lernziele und Kompetenzen	<p>Fach- Lern- bzw. Methodenkompetenz</p> <ul style="list-style-type: none"> - Wissen: Die Studierenden lernen grundlegende Repräsentationsformalismen und Algorithmen der Künstlichen Intelligenz kennen. - Anwenden: Die Konzepte werden an Beispielen aus der realen Welt angewandt (Übungsaufgaben). - Analyse: Die Studierenden lernen über die Modellierung in der Maschine menschliche Intelligenzleistungen besser einzuschätzen. <p>Sozialkompetenz</p> <ul style="list-style-type: none"> - Die Studierenden arbeiten in Kleingruppen zusammen um kleine Projekte zu bewältigen. <p>Inhalte:</p> <ul style="list-style-type: none"> • Inferenz unter Unsicherheit • Bayessche Netzwerke • Rationale Entscheidungstheorie (MDPs and POMDPs) • Machinelles Lernen und Neuronale Netzwerke • Verarbeitung Natürlicher Sprache <p>---</p> <p>This course covers the foundations of Artificial Intelligence (AI), in particular reasoning under uncertainty, machine learning and (if there is time) natural language understanding.</p> <p>This course builds on the course Artificial Intelligence I from the preceding winter semester and continues it.</p> <p>Learning Goals and Competencies</p> <p>Technical, Learning, and Method Competencies</p> <ul style="list-style-type: none"> • Knowledge: The students learn foundational representations and algorithms in AI. • Application: The concepts learned are applied to examples from the real world (homeworks). • Analysis: By modeling human cognitive abilities, students learn to assess and understand human intelligence better. • Social Competences: Students work in small groups to solve the and machine learning challenge/competition. <p>Contents:</p> <ul style="list-style-type: none"> • Inference under Uncertainty • Bayesian Networks

		<ul style="list-style-type: none"> • Rational Decision Theory (MDPs and POMDPs) • Machine Learning and Neural Networks • Natural Language Processing
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 6
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 Es werden 8-12 Übungsaufgaben gestellt, in denen Bonuspunkte gesammelt werden können. Für das Bestehen des Moduls muss nur die 90-minütige schriftliche Klausur bestanden werden.
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Es werden 8-12 Übungsaufgaben gestellt, in denen Bonuspunkte gesammelt werden können. Für das Bestehen des Moduls muss nur die 90-minütige schriftliche Klausur bestanden werden.
11	Berechnung der Modulnote	Klausur (100%) Die Note ergibt sich hauptsächlich aus der 90-minütigen schriftlichen Klausur. Bei Bestehen der Klausur kann die Note um bis zu 10% durch Punkte aus den Übungsaufgaben aufgebessert werden.
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	<p>Die Vorlesung folgt weitgehend dem Buch</p> <p>Stuart Russell und Peter Norvig: Artificial Intelligence: A Modern Approach. Prentice Hall, 3rd edition, 2009.</p> <p>Deutsche Ausgabe:</p> <p>Stuart Russell und Peter Norvig: Künstliche Intelligenz: Ein Moderner Ansatz. Pearson-Studium, 2004 (Übersetzung der 2. Auflage).</p> <p>ISBN: 978-3-8273-7089-1.</p> <p>Literature</p> <p>The course follows the following textbook: Stuart Russell and Peter Norvig: Artificial Intelligence: A Modern Approach. Prentice Hall, 3rd edition, 2009.</p>

1	Modulbezeichnung 740665	Parallele Systeme (Vorlesung mit erweiterten Übungen) Parallel systems with extended exercises	7,5 ECTS
2	Lehrveranstaltungen	Vorlesung: Parallele Systeme (2 SWS) Übung: Übung zu Parallele Systeme (2 SWS) Übung: Erweiterte Übungen zu Parallele Systeme (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Teich Frank Hannig Dominik Walter Batuhan Sesli	

4	Modulverantwortliche/r	Joachim Falk Frank Hannig Prof. Dr.-Ing. Jürgen Teich
5	Inhalt	<p>Aktuelle PCs verfügen über Mehrkernprozessoren und Grafikkarten, die wiederum aus hunderten von einfachen Prozessoren bestehen können. Hierdurch wird ein hohes Maß an nebenläufiger Datenverarbeitung möglich, welche bis vor einigen Jahren nur in Großrechnern erreicht werden konnte. Die effiziente Ausnutzung dieser Parallelität bedarf allerdings mehr als nur mehrerer Prozessoren, insbesondere muss das zu lösende Problem Parallelverarbeitung erlauben. In dieser Vorlesung werden Eigenschaften unterschiedlicher paralleler Rechnerarchitekturen und Metriken zu deren Beurteilung behandelt. Weiterhin werden Modelle und Sprachen zum Programmieren paralleler Rechner eingeführt. Neben der Programmierung von allgemeinen Parallelrechnern werden Entwurfsmethoden (CAD) vorgestellt, wie man ausgehend von einer algorithmischen Problemstellung ein massiv paralleles Rechenfeld in VLSI herleiten kann, das genau dieses Problem optimal parallel berechnet. Solche Schaltungen spielen auf der Bit- bzw. Wortebene eine dominante Rolle (Arithmetik) sowie bei Problemen der Signal- und Bildverarbeitung (z.B. Filter).</p> <p>Im Einzelnen werden behandelt:</p> <ol style="list-style-type: none"> 1) Theorie der Parallelität (parallele Computermodelle, parallele Spezifikationsformen und -sprachen, Performanzmodelle und -berechnung) 2) Klassifikation paralleler und skalierbarer Rechnerarchitekturen (Multiprozessoren und Multicomputer, Vektorrechner, Datenflussmaschinen, VLSI-Rechenfelder) 3) Programmierbare System-on-Chip (SoC) und Mehrkern-Architekturen (Grafik-Prozessoren, Cell, etc.) 4) Programmierung paralleler Rechner (Sprachen und Modelle, Entwurfsmethoden und Compiler, Optimierung) 5) Massive Parallelität: Vom Algorithmus zur Schaltung 6) Praktische Übungen mit rechnergestützten Werkzeugen <p><i>Today's PCs consist of multi-core processors and graphics cards that again comprise hundreds to thousands of simple processors. As a</i></p>

		<p>result of this, a very high degree of parallel data processing becomes possible, which was subjected to supercomputers a couple of years ago. The efficient exploitation of parallel processing requires not only multiple processors but also parallelism inherent in the problem to be solved. In this lecture, properties of different parallel computer architectures and corresponding quality metrics are examined. Further, models and parallel programming languages are introduced. In addition to programming general parallel computers, design methods (CAD) are presented that systematically transform an algorithmic problem description into a massive parallel processor array (VLSI), which can optimally execute the given problem in parallel. Such highly parallel circuits play an essential role at the bit level and circuit level (arithmetics) as well as in the case of signal processing and image processing (e.g., filter).</p> <p>In detail, the following topics are covered:</p> <ol style="list-style-type: none"> 1) Theory of parallelism (parallel models of computation, parallel specification and parallel languages, performance models) 2) Classification of parallel and scalable computer architectures (multi-processors and multi-computers, vector computers, data-flow machines, VLSI processor arrays) 3) Programmable System-on-Chip (SoC) and multi-core architectures (graphics processors, Cell, etc.) 4) Programming of parallel computers (languages and models, design methods and compiler, optimization) 5) Massive parallelism: From algorithm to circuit 6) Practical training with computer-aided design tools
6	<p>Lernziele und Kompetenzen</p>	<p>Schwerpunkt der Vorlesung ist die Vermittlung von Grundlagen der parallelen Datenverarbeitung.</p> <p><i>The focus of this lecture are foundations of parallel data processing.</i></p> <p>Fachkompetenz - Verstehen</p> <ul style="list-style-type: none"> • Die Studierenden verstehen grundlegende Konzepte der parallelen Datenverarbeitung, sowohl theoretischer Art anhand von Modellen, als auch an Architekturbeispielen. The students become familiar with the fundamentals of parallel data processing, theoretic in the form of models as well as by architecture examples. <p>Fachkompetenz - Anwenden</p> <ul style="list-style-type: none"> • Die Studierenden setzen sich mit modernen eingebetteten parallelen Ein-Chip-Architekturen auseinander. The students get familiar with modern embedded parallel system-on-chip architectures. • Die Studierenden wenden grundlegende Performanzmodelle und Parallelisierungstechniken zur Analyse und Optimierung von parallelen Algorithmen und Architekturen an. The students exercise basic performance models and parallelization techniques for the analysis and optimization of parallel algorithms and architectures.

		<ul style="list-style-type: none"> Die Studierenden setzen die Modellierung und den Entwurf von massiv-parallelen Prozessorfeldern in konkreten Aufgaben selbstständig um. In concrete tasks, the students apply independently the modeling and the design of massively parallel processors arrays. Die Studierenden wenden das erlernte Wissen in den erweiterten Übungen vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls an. The students apply their learned knowledge in hands-on computer exercises on-site at the chair's computer workstations.
7	Voraussetzungen für die Teilnahme	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Parallele Systeme“ aus.
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>Klausur (90 Minuten)</p> <ul style="list-style-type: none"> Prüfung und erfolgreiche Bearbeitung aller Übungsaufgaben in den erweiterten Übungen (verpflichtend, vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls). Ein Wechsel der Prüfungsform von einer Klausur zu einer mündlichen Prüfung ist in Ausnahmefällen (siehe § 16 ABMPO/TechFak) auch nach Semesterbeginn noch möglich. In diesem Fall werden die Studierenden spätestens zwei Wochen nach Vorlesungsbeginn informiert. Prüfungssprache: Deutsch oder Englisch. Die Unterrichts- und Prüfungssprache hängt von den Sprachkenntnissen und Präferenzen der Teilnehmerinnen und Teilnehmer ab und wird dementsprechend innerhalb der ersten zwei Wochen nach Vorlesungsbeginn festgelegt.
11	Berechnung der Modulnote	<p>Klausur (100%)</p> <p>Die Modulnote ergibt sich aus der Prüfungsleistung.</p>
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	<p>Präsenzzeit: 90 h</p> <p>Eigenstudium: 135 h</p>
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch oder Englisch
16	Literaturhinweise	<p>Weitere Informationen:</p> <p>https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/parallele-systeme</p>

1	Modulbezeichnung 768903	Computational Optics CE and MAOT Computational optics CE and MAOT	7,5 ECTS
2	Lehrveranstaltungen	Übung: Computer Exercises CompOpt CE+MAOT (2 SWS) Vorlesung mit Übung: Computational Optics CE & MAOT (2 SWS)	- 7,5 ECTS
3	Lehrende	Souryadeep Saha Prof. Dr. Christoph Pflaum	

4	Modulverantwortliche/r	Prof. Dr. Christoph Pflaum
5	Inhalt	Simulation optischer Wellen Finite-Differenzen-Methode zur Lösung der Maxwell'schen Gleichungen Strahl-Propagations-Methoden Ratengleichungen für Photonen Anwendung im Bereich der Simulation von Lasern und Dünnschichtsolarzellen
6	Lernziele und Kompetenzen	Fachkompetenz Anwenden Anwendung unterschiedlicher Simulationstechniken in der Optik Analysieren Analyse der Stabilität von Simulationstechniken Erschaffen Entwicklung von Software zur Simulation von optischen Wellen
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 6
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	mündlich Die Prüfungsdauer ist 30 Minuten.
11	Berechnung der Modulnote	mündlich (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 75 h Eigenstudium: 150 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	

1	Modulbezeichnung 773774	Eingebettete Systeme (Vorlesung mit erweiterten Übungen) Embedded systems	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Joachim Falk Prof. Dr.-Ing. Jürgen Teich	
5	Inhalt	<p>Das Modul, Eingebettete Systeme mit erweiterter Übung, thematisiert den Entwurf und die Implementierung eingebetteter Systeme unter Einsatz formaler Methoden und rechnergestützter Entwurfsverfahren. Unter eingebetteten Systemen versteht man Rechensysteme, die auf einen Anwendungsbereich zugeschnitten (z.B. mobile Kommunikationsgeräte, Chipkartensysteme, Industriesteuerungen, Unterhaltungselektronik, Medizintechnik) und in einen technischen Kontext eingebunden sind. Das große Interesse am systematischen Entwurf von heterogenen eingebetteten Systemen ist verursacht durch die steigende Vielfalt und Komplexität von Anwendungen für eingebettete Systeme, die Notwendigkeit, Entwurfs- und Testkosten zu senken sowie durch Fortschritte in Schlüsseltechnologien (Mikroelektronik, formale Methoden).</p> <p><i>The focus of this module is the design and implementation of embedded systems using formal methods and computer-aided design techniques. Embedded systems are computing systems tailored for a particular application (e.g., mobile communication devices, smart card systems, industrial control, consumer electronics, medical technology) and integrated into a technical context. The keen interest in the systematic design of heterogeneous embedded systems is driven by the increasing diversity and complexity of embedded system applications, the need to reduce design and test costs, and advances in key technologies (microelectronics, formal methods).</i></p>	
6	Lernziele und Kompetenzen	<p>Fachkompetenz - Wissen</p> <ul style="list-style-type: none"> • Die Studierenden setzen sich mit einem aktuellen Forschungsgebiet auseinander. The students deal with a current field of research. • In den erweiterten Übungen lernen die Studierenden aktuelle Entwurfswerkzeuge für die Architektursynthese (Hardware) und Softwaresynthese vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls kennen. In the extended exercises, the students learn about current design tools for architecture synthesis (hardware) and software synthesis on-site at the chair's computer workstations. <p>Fachkompetenz - Verstehen</p>	

		<ul style="list-style-type: none"> • Die Studierenden verstehen grundlegende Konzepte des Entwurfs eingebetteter Systeme. The students become familiar with the fundamental concepts of designing of embedded systems. <p>Fachkompetenz - Anwenden</p> <ul style="list-style-type: none"> • Die Studierenden wenden grundlegende Algorithmen an zur Analyse und Optimierung von Hardware-Architekturen und Echtzeit-Softwaresystemen. The students apply basic algorithms to analyze and optimize hardware architectures and real-time software systems. • Die Studierenden erfassen den Hardware/Software-Entwurf von Systemen mit harten Beschränkungen. The students understand the hardware/software design of hard-constrained systems. • Die Studierenden wenden aktuelle Entwurfswerkzeuge, die auf den Rechnerarbeitsplätzen des Lehrstuhls installiert sind, an, um damit die Aufgaben der erweiterten Übungen unter Anleitung zu lösen. The students apply current design tools installed on the chair's computer workstations to solve the tasks of the extended exercises with the help of instructions. <p>Sozialkompetenz</p> <ul style="list-style-type: none"> • Die Studierenden lernen aktuelle Entwurfswerkzeuge für die Architektursynthese (Hardware) und Softwaresynthese kennen bei der kooperativen Bearbeitung der erweiterten Übung in Gruppen. The students learn about current design tools for architecture synthesis (hardware) and software synthesis by processing the extended exercises in groups cooperatively.
7	<p>Voraussetzungen für die Teilnahme</p>	<p>Die Auswahl dieses Moduls schließt die Auswahl der Module „Eingebettete Systeme“ und „Eingebettete Systeme (Vorlesung mit Übungen)“ aus. <i>Selecting this module excludes the selection of the modules “Embedded Systems” and “Embedded Systems (Lecture with Exercises)”.</i></p> <p>Organisatorisches:</p> <ul style="list-style-type: none"> • Die Vorlesung erfolgt in deutscher Sprache. Zusätzlich stehen Folien und Vorlesungsaufzeichnungen in englischer Sprache zur Verfügung. • Die Übungen werden sowohl auf Deutsch als auch auf Englisch angeboten. • Studierende können die Prüfung wahlweise auf Deutsch oder Englisch ablegen. <p>Organizational:</p> <ul style="list-style-type: none"> • The lecture is given in German. Slides and lecture recordings are also provided in English. • German as well as English exercises are offered. • Students can choose between taking the exam either in German or English.

8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Klausur (90 min) und erfolgreiche Bearbeitung aller Übungsaufgaben in den erweiterten Übungen (verpflichtend, vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls). <i>Written exam (90 min) and successful completion of all exercises in the extended exercises (mandatory, on site at the computer workstations of the chair).</i>
11	Berechnung der Modulnote	Klausur (100%) Die Modulnote ergibt sich aus der Klausurnote. <i>The grade of the module is the grade of the exam.</i>
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	Empfohlenes Buch zur Begleitung und Vertiefung: <ul style="list-style-type: none"> Teich J., Haubelt C.: "Digitale Hardware/Software-Systeme: Synthese und Optimierung", Springer-Verlag, 2007, ISBN: 978-3-540-46822-6 Weitere Informationen: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/eingebettete-systeme/

1	Modulbezeichnung 796399	Geometric Modeling Geometric modeling	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Dr. Roberto Grosso Prof. Dr. Marc Stamminger	
5	Inhalt	<p>Das Modul beschäftigt sich mit Methoden zur Modellierung dreidimensionaler Oberflächen. Typische Einsatzgebiete sind der rechnerunterstützte Entwurf (CAD, z.B. im Automobil- oder Flugzeugbau), die Rekonstruktion von Flächen aus Sensordaten oder die Konstruktion glatter Interpolationsflächen. Behandelt werden u.a. folgende Themen:</p> <ul style="list-style-type: none"> • Polynomkurven • Bezierkurven, rationale Bezierkurven • B-Splines • Tensorproduktflächen • Bezier-Dreiecksflächen • polygonale Flächen • Subdivision-Verfahren <p>This module is concerned with different aspects of modelling three-dimensional curves and surfaces. Typical areas of application are computer-aided design (CAD), reconstruction of surfaces from sensor data (reverse engineering) and construction of smooth interpolants. The lecture covers the following topics:</p> <ul style="list-style-type: none"> • polynomial curves • Bézier curves, rational Bézier curves • B-splines • tensor product surfaces • triangular Bézier surfaces • polyhedral surfaces 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • erklären die Begriffe Polynomkurve, Bezierkurven und B-Splines • klassifizieren und veranschaulichen die unterschiedlichen Auswertungs- und Subdivision-Verfahren für Bezier-Kurven und B-Splines • veranschaulichen und ermitteln die Eigenschaften von Bezierkurven, rationalen Bezierkurven und B-Splines • beschreiben Tensorproduktflächen und skizzieren Auswertungsalgorithmen • erklären polygonale Flächen und Subdivision-Verfahren und veranschaulichen ihre Unterschiede und Eigenschaften • lernen gängige Datenstrukturen zur Darstellung polygonaler Flächen kennen 	

		<ul style="list-style-type: none"> • wenden die Verfahren der Geometrischen Modellierung an unterschiedliche Beispiele an • berechnen Bezierkurven und B-Splines und analysieren ihre Eigenschaften • führen Subdivision-Verfahren für Kurven und Flächen aus und analysieren ihre Eigenschaften • analysieren und evaluieren Stetigkeitseigenschaften von Bezierkurven und B-Splines • analysieren und evaluieren Stetigkeitseigenschaften von Subdivision-Surface • implementieren alle Verfahren für Kurven und Flächen in JavaScript <p>Educational objectives and skills: The students</p> <ul style="list-style-type: none"> • explain the terms polynomial curve, Bezier curves and B-splines • classify and illustrate the different evaluation and subdivision methods for Bezier curves and B-splines • illustrate and determine the properties of Bezier curves, rational Bezier curves and B-splines • describe tensor product surfaces and outline evaluation algorithms • explain polygonal surfaces and subdivision methods and illustrate their differences and properties • learn about common data structures for representing polygonal surfaces • apply geometric modeling methods to different examples • calculate Bezier curves and B-splines and analyze their properties • carry out subdivision methods for curves and surfaces and analyze their properties • analyze and evaluate continuity properties of Bezier and B-spline curves • analyze and evaluate continuity properties of subdivision surfaces • implement all methods for curves and surfaces in JavaScript
7	<p>Voraussetzungen für die Teilnahme</p>	<p>Voraussetzungen (empfohlen, aber nicht erforderlich)</p> <ul style="list-style-type: none"> • Computergrafik • Vektorrechnung, lineare Algebra • Programmierkenntnisse sind erforderlich. Für die Programmieraufgabe und die Abschlussprüfung sind Kenntnisse in JavaScript erforderlich. <p>Prerequisites (Recommended but not)</p> <ul style="list-style-type: none"> • computer graphics • vector calculus, linear algebra

		<ul style="list-style-type: none"> Programming skills are required. The programming assignment and the final examination require some knowledge of JavaScript.
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>elektronische Prüfung mit MultipleChoice</p> <p>Die Klausur ist eine elektronische Präsenzprüfung. Die Klausur kann einen Multiple-Choice-Teil enthalten. Um die Klausur zu bestehen, müssen zusätzlich folgende Punkte beachtet werden:</p> <ul style="list-style-type: none"> Die elektronische Klausur besteht aus Theorie-, Praxis- und Programmieraufgaben. Außerdem müssen 50% der möglichen Gesamtpunktzahl erreicht werden. Die Klausuren sind so formuliert, dass es sehr schwierig ist, sie nur mit Theoriekenntnissen und praktischen Aufgaben zu bestehen. Die Lösung von Programmieraufgaben ist ein wesentlicher Bestandteil der in diesem Modul vermittelten Kompetenzen.
11	Berechnung der Modulnote	elektronische Prüfung mit MultipleChoice (100%) Die Note des Moduls ergibt sich aus der Note der elektronischen Prüfung (100 %).
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> Hoschek, Lasser: Grundlagen der Geometrischen Datenverarbeitung Farin: Kurven und Flächen im Computer Aided Geometric Design de Boor: A Practical Guide to Splines Bartels, Beatty, Barsky: Splines for Use in Computer Graphics and Geometric Modeling Abramowski, Müller: Geometrisches Modellieren

1	Modulbezeichnung 798810	Rechnerarchitektur Computer architecture	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Dietmar Fey	
5	Inhalt	<p>Die Vorlesung baut auf die in den Grundlagen der Rechnerarchitektur und -organisation vermittelten Inhalte auf und setzt diese mit weiterführenden Themen fort. Es werden zunächst grundlegende fortgeschrittene Techniken bei Pipelineverarbeitung und Cachezugriffen in modernen Prozessoren und Parallelrechnern behandelt. Ferner wird die Architektur von Spezialprozessoren, z.B. DSPs und Embedded Prozessoren behandelt. Es wird aufgezeigt, wie diese Techniken in konkreten Architekturen (Intel Nehalem, GPGPU, Cell BE, TMS320 DSP, Embedded Prozessor ZPU) verwendet werden. Zur Vorlesung wird eine Tafelübung angeboten. Mit erfolgreicher mündlicher Prüfung können 5 ECTS erworben werden. In den Tafelübungen werden die in der Vorlesung vermittelten Techniken durch zu lösende Aufgaben vertieft. In der Rechnerübung soll u.a. ein einfacher Vielkern-Prozessor auf Basis des ZPU-Prozessors mit Simulationswerkzeugen aufgebaut werden. Im Einzelnen werden folgende Themen behandelt:</p> <ul style="list-style-type: none"> • Organisationsaspekte von CISC und RISC-Prozessoren • Behandlung von Hazards in Pipelines • Fortgeschrittene Techniken der dynamischen Sprungvorhersage • Fortgeschritten Cachetechniken, Cache-Kohärenz • Ausnutzen von Cacheeffekten • Architekturen von Digitalen Signalprozessoren • Architekturen homogener und heterogener Multikern-Prozessoren (Intel Corei7, Nvidia GPUs, RISC-V) • Architektur von Parallelrechnern (Clusterrechner, Superrechner) • Effiziente Hardware-nahe Programmierung von Multikern-Prozessoren (OpenMP, SSE, CUDA) • Leistungsmodellierung und -analyse von Multikern-Prozessoren (Roofline-Modell) 	
6	Lernziele und Kompetenzen	<p>Fachkompetenz Wissen Lernende können Wissen abrufen und wiedergeben. Sie kennen konkrete Einzelheiten wie Begriffe, Definitionen, Fakten, und Abläufe in einem Prozessor darlegen.</p> <p>Verstehen Lernende können Beispiele für Rechnerarchitekturen anführen, sie sind in der Lage, Schaubilder von Prozessoren zu interpretieren und die Abläufe in eigenen Worten zu beschreiben.</p>	

		<p>Anwenden Lernende können beim Erstellen eigener Programme durch Transfer des Wissens über Interna von Prozessorarchitekturen Optimierungen hinsichtlich des Laufzeitverhaltens vornehmen.</p> <p>Analysieren Lernende können zwischen verschiedenen Varianten von Lösungen einer Prozessorarchitektur klassifizieren, die Gründe für durchgeführte Entwurfsentscheidungen erschließen, Unterscheide gegenüberstellen und gegeneinander bewerten.</p> <p>Lern- bzw. Methodenkompetenz Lernende erwerben die Fähigkeit selbstständig Testprogramme zum Bewerten der Leistungsfähigkeit eines Prozessors zu erstellen.</p>
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Variabel
11	Berechnung der Modulnote	Variabel (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> • Patterson/Hennessy: Computer Organization und Design • Hennessy/Patterson: Computer Architecture - A Quantitative Approach • Stallings: Computer Organization and Architecture • Martin: Rechnerarchitekturen

1	Modulbezeichnung 837601	Mikromechanik Micromechanics	2,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	apl. Prof. Dr. Julia Mergheim	
5	Inhalt	<ul style="list-style-type: none"> • Grundlagen der linearen Kontinuumsmechanik • Elastizität • mean-field approaches und variational bounding methods • numerische Homogenisierung • FE² Methode • weitere Multiskalen-Methoden 	
6	Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • sind vertraut mit den theoretischen Grundlagen der Mikromechanik • können analytische Homogenisierungsmethoden einsetzen • kennen geeignete Homogenisierungsverfahren 	
7	Voraussetzungen für die Teilnahme	Empfohlen: Grundkenntnisse in Kontinuumsmechanik Alle Informationen zum Ablauf der Lehrveranstaltung werden über den StudOn-Kurs kommuniziert. Deshalb bitten wir Sie, sich unter https://www.studon.fau.de/cat5282.html einzuschreiben. Der Beitritt ist nicht, wie sonst üblich, passwortgeschützt, sondern erfolgt nach Bestätigung durch den Dozenten. Dies geschieht mitunter nicht umgehend, aber rechtzeitig vor dem ersten Termin. Wir bitten um Ihr Verständnis.	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	mündlich <ul style="list-style-type: none"> • Mikromechanik (Prüfungsnummer: 837601) • Prüfungsleistung, mündliche Prüfung, Dauer (in Minuten): 30, benotet 	
11	Berechnung der Modulnote	mündlich (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 30 h Eigenstudium: 45 h	
14	Dauer des Moduls	1 Semester	

15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 894856	Künstliche Intelligenz I Artificial intelligence I	7,5 ECTS
2	Lehrveranstaltungen	Übung: Übungen zu Künstliche Intelligenz I (2 SWS)	-
3	Lehrende	apl. Prof. Dr. Florian Rabe	

4	Modulverantwortliche/r	Prof. Dr. Michael Kohlhase
5	Inhalt	<p>Dieses Modul beschäftigt sich mit den Grundlagen der Künstlichen Intelligenz (KI), insbesondere formale Wissensrepräsentation, Heuristische Suche, Automatisches Planen und Schliessen unter Unsicherheit.</p> <p>---</p> <p>This module covers the foundations of Artificial Intelligence (AI), in particular symbolic techniques based on search and inference.</p>
6	Lernziele und Kompetenzen	<ul style="list-style-type: none"> - Wissen: Die Studierenden lernen grundlegende Repräsentationsformalismen und Algorithmen der Künstlichen Intelligenz kennen. - Anwenden: Die Konzepte werden an Beispielen aus der realen Welt angewandt (Übungsaufgaben). - Analyse: Die Studierenden lernen die über die modellierung in der Maschine menschliche Intelligenzleistungen besser einzuschätzen. <p>Sozialkompetenz</p> <ul style="list-style-type: none"> - Die Studierenden arbeiten in Kleingruppen zusammen um kleine Projekte zu bewältigen <p>*Inhalt*:</p> <ul style="list-style-type: none"> - Agentenmodelle als Grundlage der Künstlichen Intelligenz - Logisches Programmieren in Prolog - Heuristische Suche als Methode zur Problemlösung - Zwei-Agenten-Suche (automatisierung von Brettspielen) mittels heuristischer Suche - Constraint Solving/Propagation - Logische Sprachen für die Wissensrepräsentation - Inferenz and Automatisiertes Theorembeweisen (DPLL-Varianten und PL1)_ - Classisches Planen - Planen und Agieren in der wirklichen Welt. <p>---</p> <p>Technical, Learning, and Method Competencies</p> <ul style="list-style-type: none"> - Knowledge: The students learn foundational representations and algorithms in AI. - Application: The concepts learned are applied to examples from the real world (homeworks). - Analysis: By modeling human cognitive abilities, students learn to assess and understand human intelligence better. - Social Competences: Students work in small groups to solve an AI game-play challenge/competition (Kalah). <p>Contents: Foundations of symbolic AI, in particular:</p> <ul style="list-style-type: none"> - Agent Models as foundation of AI - Logic Programming in Prolog

		<ul style="list-style-type: none"> - Heuristic Search as a method for problem solving - Adversarial Search (automating board games) via heuristic search - Constraint Solving/Propagation - Logical Languages for knowledge representation - Inference and automated theorem proving - Classical Planning - Planning and Acting in the real world.
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Es werden 8-12 Übungsaufgaben gestellt, in denen Bonuspunkte gesammelt werden können. Für das Bestehen des Moduls muss nur die 90-minütige schriftliche Klausur bestanden werden.
11	Berechnung der Modulnote	Klausur (100%) Die Note ergibt sich hauptsächlich aus der 90-minütigen schriftlichen Klausur. Bei Bestehen der Klausur kann die Note um bis zu 10% durch Punkte aus den Übungsaufgaben aufge bessert werden.
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	<p>Die Vorlesung folgt weitgehend dem Buch</p> <p>Stuart Russell und Peter Norvig: Artificial Intelligence: A Modern Approach. Prentice Hall, 3rd edition, 2009.</p> <p>Deutsche Ausgabe:</p> <p>Stuart Russell und Peter Norvig: Künstliche Intelligenz: Ein Moderner Ansatz. Pearson-Studium, 2004 (Übersetzung der 2. Auflage). ISBN: 978-3-8273-7089-1.</p>

1	Modulbezeichnung 43371	Interactive Computer Graphics Interactive computer graphics	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Interactive Computer Graphics (2 SWS) Übung: Tutorials to Interactive Computer Graphics (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Marc Stamminger Laura Fink	

4	Modulverantwortliche/r	Prof. Dr. Marc Stamminger	
5	Inhalt	<p>In dem Modul werden GPUs und dafür maßgeschneiderte Algorithmen behandelt:</p> <ul style="list-style-type: none"> • Architektur von GPUs und Echtzeit-Rendering-Pipeline • Deferred Shading und Anti-Aliasing-Verfahren • Simulation von Umgebungsbeleuchtung • Verfahren zur Generierung von Schatten • Level-of-Detail-Verfahren zur Darstellung komplexer Szenen • Animation von Objekten 	
6	Lernziele und Kompetenzen	<p>Fachkompetenz Verstehen Lernende können verschiedene in der Vorlesung behandelte Verfahren der interaktiven Computergraphik mit eigenen Worten erklären. Anwenden Lernende können Zusammenhänge zwischen den in der Vorlesung behandelten Verfahren der interaktiven Computergraphik erkennen, und Ideen auf neue Anwendungen übertragen. Analysieren Lernende können Unterschiede und Ähnlichkeiten der in der Vorlesung behandelten Verfahren erkennen und daraus neue Lösungen entwickeln.</p>	
7	Voraussetzungen für die Teilnahme	Die Übungen setzen Kenntnisse in C/C++ voraus.	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Variabel Übungsleistung	
11	Berechnung der Modulnote	Variabel (100%) Übungsleistung (bestanden/nicht bestanden)	
12	Turnus des Angebots	nur im Sommersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Deutsch Englisch	

1	Modulbezeichnung 43374	Computer Graphics Deluxe	7,5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Marc Stamminger
5	Inhalt	<p>Die Vorlesung gibt eine Einführung in die Computergraphik:</p> <ul style="list-style-type: none"> • Graphik Pipeline • Clipping • 3D Transformationen • Hierarchische Display Strukturen • Perspektive und Projektionen • Sichtbarkeitsbetrachtungen • Rastergraphik und Scankonvertierung • Farbmodelle • Lokale und globale Beleuchtungsmodelle • Schattierungsverfahren • Ray Tracing und Radiosity • Schatten und Texturen <p>Contents: This lecture covers the following aspects of Computer Graphics:</p> <ul style="list-style-type: none"> • graphics pipeline • clipping • 3D transformations • hierarchical display structures • perspective transformations and projections • visibility determination • raster graphics and scan conversion • color models • local and global illumination models • shading models • ray tracing and radiosity • shadows and textures
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • geben die unterschiedlichen Schritte der Graphik Pipeline wieder • erklären die Funktionsweise der Clippingalgorithmen für Linien und Polygone • beschreiben, charakterisieren und berechnen affine und perspektivische Transformationen in 3D und veranschaulichen die allgemeine Form der Transformationsmatrix in homogener Koordinaten • skizzieren die Verfahren zur Tiefe- und Sichtbarkeitsberechnung • vergleichen die unterschiedlichen Farbmodelle der Computergraphik

		<ul style="list-style-type: none"> • illustrieren und untersuchen die Datenstrukturen zur Beschreibung virtueller 3D Modelle und komplexer Szenen • erläutern die Funktionsweise der Rasterisierung und Scankonvertierung in der Graphikpipeline • implementieren 3D Transformationen mithilfe der Programmiersprache C++ und der graphischen Bibliothek OpenGL • Implementieren Beleuchtungsmodelle und Texturierung von virtuellen 3D Objekten mithilfe der Programmiersprachen OpenGL und GLSL • lösen Aufgaben zu Beleuchtung und Texturierung von 3D virtuellen Modellen • klassifizieren Schattierungsverfahren • bestimmen den Unterschied zwischen lokaler und globaler Beleuchtung und formulieren Algorithmen für Ray Tracing und Radiosity <p>*Educational objectives and skills:*</p> <p>Students should be able to</p> <ul style="list-style-type: none"> • describe the processing steps in the graphics pipeline • explain clipping algorithms for lines and polygons • explain, characterize and compute affine and perspective transformations in 2D and 3D, and provide an intuitive description of the general form of corresponding transformation matrices in homogeneous coordinates • depict techniques to compute depth, occlusion and visibility • compare the different color models • describe data structures to represent 3D virtual models and complex scenes • explain the algorithms for rasterization and scan conversion • solve problems with shading and texturing of 3D virtual models • classify different shadowing techniques • explain the difference between local and global illumination techniques and formulate algorithms for ray tracing and radiosity
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 Sowohl die Computer Graphics Basic Tutorials als auch die Computer Graphics Advanced Tutorials bestehen aus 10 wöchentlichen Aufgabenblättern mit kleinen Programmieraufgaben.
10	Studien- und Prüfungsleistungen	Übungsleistung Übungsleistung Zum Bestehen des Moduls müssen 50% der Punkte in den Übungen erreicht und die Prüfung bestanden werden. Die Modulnote ergibt sich zu 100% aus der mündlichen Prüfung.

11	Berechnung der Modulnote	Übungsleistung (bestanden/nicht bestanden) Übungsleistung (bestanden/nicht bestanden)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> • P. Shirley: Fundamentals of Computer Graphics. AK Peters Ltd., 2002 • Hearn, M. P. Baker: Computer Graphics with OpenGL. Pearson • Foley, van Dam, Feiner, Hughes: Computer Graphics - Principles and Practice • Rauber: Algorithmen der Computergraphik • Bungartz, Griebel, Zenger: Einführung in die Computergraphik • Encarnaç�o, Strasser, Klein: Computer Graphics

1	Modulbezeichnung 43822	Computer Graphics Computer graphics	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Marc Stamminger	
5	Inhalt	<p>Die Vorlesung gibt eine Einführung in die Computergraphik:</p> <ul style="list-style-type: none"> • Graphik Pipeline • Clipping • 3D Transformationen • Hierarchische Display Strukturen • Perspektive und Projektionen • Sichtbarkeitsbetrachtungen • Rastergraphik und Scankonvertierung • Farbmodelle • Lokale und globale Beleuchtungsmodelle • Schattierungsverfahren • Ray Tracing und Radiosity • Schatten und Texturen <p>Contents: This lecture covers the following aspects of Computer Graphics:</p> <ul style="list-style-type: none"> • graphics pipeline • clipping • 3D transformations • hierarchical display structures • perspective transformations and projections • visibility determination • raster graphics and scan conversion • color models • local and global illumination models • shading models • ray tracing and radiosity • shadows and textures 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • geben die unterschiedlichen Schritte der Graphik Pipeline wieder • erklären die Funktionsweise der Clippingalgorithmen für Linien und Polygone • beschreiben, charakterisieren und berechnen affine und perspektivische Transformationen in 3D und veranschaulichen die allgemeine Form der Transformationsmatrix in homogener Koordinaten • skizzieren die Verfahren zur Tiefe- und Sichtbarkeitsberechnung 	

		<ul style="list-style-type: none"> • vergleichen die unterschiedlichen Farbmodelle der Computergraphik • illustrieren und untersuchen die Datenstrukturen zur Beschreibung virtueller 3D Modelle und komplexer Szenen • erläutern die Funktionsweise der Rasterisierung und Scankonvertierung in der Graphikpipeline • lösen Aufgaben zu Beleuchtung und Texturierung von 3D virtuellen Modellen • klassifizieren Schattierungsverfahren • bestimmen den Unterschied zwischen lokaler und globaler Beleuchtung und formulieren Algorithmen für Ray Tracing und Radiosity <p>*Educational objectives and skills:*</p> <p>Students should be able to</p> <ul style="list-style-type: none"> • describe the processing steps in the graphics pipeline • explain clipping algorithms for lines and polygons • explain, characterize and compute affine and perspective transformations in 2D and 3D, and provide an intuitive description of the general form of corresponding transformation matrices in homogeneous coordinates • depict techniques to compute depth, occlusion and visibility • compare the different color models • describe data structures to represent 3D virtual models and complex scenes • explain the algorithms for rasterization and scan conversion • solve problems with shading and texturing of 3D virtual models • classify different shadowing techniques • explain the difference between local and global illumination techniques and formulate algorithms for ray tracing and radiosity
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>Übungsleistung Klausur (60 Minuten) Die Übungen ("Computer Graphics Basic Tutorials") bestehen aus insgesamt 10 wöchentlichen Aufgabenblättern mit kleinen Programmieraufgaben.</p> <p>The exercises ("Computer Graphics Basic Tutorials") consist of weekly worksheets (10 worksheets in total) with small programming tasks.</p>
11	Berechnung der Modulnote	Übungsleistung (bestanden/nicht bestanden) Klausur (100%)

		<p>Zum Bestehen des Moduls müssen 50% der Punkte in den Übungen erreicht und die Abschlussprüfung bestanden werden. Die Modulnote ergibt sich zu 100% aus der Prüfung.</p> <p>The module is passed when 50% of the points in the exercises are reached and when the final exam is passed. The grade of the module is entirely determined by the grade in the final exam.</p>
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> • P. Shirley: Fundamentals of Computer Graphics. AK Peters Ltd., 2002 • Hearn, M. P. Baker: Computer Graphics with OpenGL. Pearson • Foley, van Dam, Feiner, Hughes: Computer Graphics - Principles and Practice • Rauber: Algorithmen der Computergraphik • Bungartz, Griebel, Zenger: Einführung in die Computergraphik • Encarnaçã, Strasser, Klein: Computer Graphics

1	Modulbezeichnung 97010	Strömungsmechanik I Fluid mechanics I	5 ECTS
2	Lehrveranstaltungen	Vorlesung: Strömungsmechanik I (2 SWS) Übung: Strömungsmechanik I - Übung (2 SWS)	5 ECTS -
3	Lehrende	Prof. Dr. Andreas Wierschem	

4	Modulverantwortliche/r	Prof. Dr. Andreas Wierschem	
5	Inhalt	<ul style="list-style-type: none"> • Charakterisierung von Fluiden • Kontinuumsannahme • Strömungskinematik: materielle und Feldbeschreibung, Bahn- und Stromlinien, materielle Zeitableitung, Relativbewegung, Reynoldssches Transporttheorem • Bilanzgleichungen: Massenbilanz, Navier-Stokes-Gleichung, integral und differentiell • Hydrostatik: Auftrieb, Druck auf Wände, kapillarer Druck, gleichmäßig beschleunigte Systeme • Grundlagen der Ähnlichkeitstheorie: Dimensionslose Kennzahlen, Grenzfälle der Navier-Stokes-Gleichung • Bernoulli-Gleichung: stationär und instationär, mit Druckverlusten und Energieaustausch. <p>Die Studierenden werden angeleitet, mit dem erhaltenen Wissen strömungsmechanische Problemstellungen zu bewerten, Lösungswege zu erarbeiten und mit eingeübten Methoden und Vorgehensweisen an Hand von Beispielen praktisch anzuwenden.</p>	
6	Lernziele und Kompetenzen	<p>Das Modul bietet eine systematische Einführung in die Strömungsmechanik.</p> <p>Die Studierenden:</p> <ul style="list-style-type: none"> • können die Bedeutung der Strömungsmechanik sowohl im Alltag als auch bei industriellen Prozessen nachvollziehen • verfügen über einen Überblick über verschiedene Regime der Strömungsmechanik und verstehen ihren Anwendungsbereich • können die erworbenen Grundkenntnisse mit eingeübten Methoden und Vorgehensweisen anhand von Beispielen in der Übung praktisch anwenden • sind fähig, strömungsmechanische Problemstellungen zu bewerten und Lösungswege anzuwenden. 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	NF Thermo- and Fluidynamics Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Klausur (120 Minuten)	

11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • J. H. Spurk, N. Aksel: Strömungslehre: Einführung in die Theorie der Strömungen , 8. Auflage, Springer-Verlag Berlin, Heidelberg, New York, 2010 • F. Durst: Grundlagen der Strömungsmechanik - Eine Einführung in die Theorie der Strömungen in Fluiden , Springer, 2006 • H. Kuhlmann: Strömungsmechanik , Pearson, 2007 • P. K. Kundu: Fluid Mechanics , 5th Ed., Academic Press, 2012 • F. M. White: Fluid Mechanics , 7th Rev. Ed., McGraw Hill, 2011 • F. A. Morrison: An Introduction to Fluid Mechanics , Cambridge University Press, 2013 • L. Böswirth: Technische Strömungslehre , 9. Auflage, Vieweg & Teubner, 2011 • W. Kümmel: Technische Strömungsmechanik - Theorie und Praxis , 3. Auflage, Teubner, 2007 • H. Sigloch: Technische Fluidmechanik , 8. Auflage, Springer, 2012 • H. Oertel Jr.: Strömungsmechanik - Grundlagen, Grundgleichungen, Lösungsmethoden, Softwarebeispiele , 6. Auflage, Vieweg & Teubner, 2011

1	Modulbezeichnung 292952	Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung) Hardware-software-co-design (Lecture with extended exercises)	7,5 ECTS
2	Lehrveranstaltungen	Übung: UE-HSCD (2 SWS) Vorlesung: Hardware-Software-Co-Design (2 SWS) Übung: Erweiterte Übungen zu Hardware-Software-Co-Design (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Muhammad Sabih Tobias Hahn Abrarul Karim Prof. Dr.-Ing. Jürgen Teich	

4	Modulverantwortliche/r	Joachim Falk Prof. Dr.-Ing. Jürgen Teich	
5	Inhalt	<p>Zahlreiche Realisierungen eingebetteter Systeme (z.B. Mobiltelefone, Faxgeräte, Industriesteuerungen) zeichnen sich durch kooperierende Hardware- und Softwarekomponenten aus. Die Popularität solcher Realisierungsformen lässt sich begründen durch 1) die steigende Vielfalt und Komplexität heterogener Systeme, 2) die Notwendigkeit, Entwurfs- und Testkosten zu senken und 3) Fortschritte in Schlüsseltechnologien (Mikroelektronik, formale Entwurfsmethoden). Zum Beispiel bieten Halbleiterhersteller kostengünstige ASICs an, die einen Mikrocontroller und benutzerspezifische Peripherie und Datenpfade auf einem Chip integrieren.</p> <p>Die Synthese solcher Systeme wirft jedoch eine Reihe neuartiger Entwurfsprobleme auf, insbesondere 1) die Frage der Auswahl von Hardware- und Softwarekomponenten, 2) die Partitionierung einer Spezifikation in Hard- und Software, 3) die automatische Synthese von Interface- und Kommunikationsstrukturen und 4) die Verifikation und Cosimulation.</p> <p>1) Überblick und Vergleich von Architekturen und Komponenten in Hardware/Software-Systemen. 2) Aufbau eines Compilers und Codeoptimierungsverfahren für Hardware und Software 3) Hardware/Software-Partitionierung (Partitionierung komplexer Systeme, Schätzungsverfahren, Performanzanalyse, Codegenerierung) 4) Interfacesynthese (Kommunikationsarten, Synchronisation, Synthese) 5) Verifikation und Cosimulation 6) Tafelübungen 7) Demonstrationen mit rechnergestützten Entwurfswerkzeugen und praktische Übungen</p> <p>Content: Numerous realisations of embedded systems (e.g. mobile phones, fax machines, industrial controls) are characterised by cooperating hardware and software components. The popularity of such realisations can be explained by 1) the increasing diversity and</p>	

		<p>complexity of heterogeneous systems, 2) the need to reduce design and testing costs, and 3) advances in key technologies (microelectronics, formal design methods). For example, semiconductor manufacturers offer low-cost ASICs that integrate a microcontroller and user-specific peripherals and data paths on a single chip.</p> <p>However, the synthesis of such systems raises a number of novel design problems, in particular 1) the issue of hardware and software component selection, 2) the partitioning of a specification into hardware and software, 3) the automatic synthesis of interface and communication structures, and 4) verification and cosimulation.</p> <ol style="list-style-type: none"> 1) Overview and comparison of architectures and components in hardware/software systems. 2) Structure of a compiler and code optimisation procedures for hardware and software. 3) Hardware/software partitioning (partitioning of complex systems, estimation procedures, performance analysis, code generation) 4) Interface synthesis (communication types, synchronisation, synthesis) 5) Verification and cosimulation 6) Blackboard exercises 7) Demonstrations with computer-aided design tools and practical exercises
6	<p>Lernziele und Kompetenzen</p>	<p>Fachkompetenz - Wissen</p> <ul style="list-style-type: none"> • Die Studierenden erhalten Einblick in ein aktuelles Forschungsgebiet. <p>Fachkompetenz - Verstehen</p> <ul style="list-style-type: none"> • Die Studierenden verstehen Grundlagen des modernen Systementwurfs. • Die Studierenden erklären Implementierungsalternativen für digitale Hardware/Software-Systeme. <p>Fachkompetenz - Anwenden</p> <ul style="list-style-type: none"> • Die Studierenden wenden grundlegende Algorithmen an, zur Analyse und Optimierung von Hardware/Software-Systemen. • Die Studierenden wenden das erlernte Wissen in den erweiterten Übungen vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls an. <p>Sozialkompetenz</p> <ul style="list-style-type: none"> • Die Studierenden benutzen aktuelle Entwurfswerkzeuge für die Spezifikation, Optimierung und Prototypisierung von Hardware/Software-Systemen bei der kooperativen Bearbeitung der erweiterten Übung in Gruppen. <p>Competences: Professional competence - Knowledge</p> <ul style="list-style-type: none"> • Students gain insight into a current field of research. <p>Professional competence - Understanding</p> <ul style="list-style-type: none"> • Students understand the basics of modern system design. • Students explain implementation alternatives for digital hardware/software systems. <p>Professional competence - Application</p>

		<ul style="list-style-type: none"> Students apply basic algorithms to analyse and optimise hardware/software systems. The students apply the knowledge they have acquired in the extended exercises on site at the computer workstations of the department. <p>Social competence</p> <ul style="list-style-type: none"> The students use current design tools for the specification, optimisation and prototyping of hardware/software systems in the cooperative processing of the extended exercise in groups.
7	Voraussetzungen für die Teilnahme	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Hardware-Software-Co-Design“ aus.
8	Einpassung in Studienverlaufsplan	Semester: 1
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten) Klausur (90 min) und erfolgreicher Bearbeitung aller Übungsaufgaben in den erweiterten Übungen (verpflichtend, vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls). Die Sprache der Klausur ist abhängig von der Wahl der Studierenden entweder Deutsch oder Englisch.
11	Berechnung der Modulnote	Klausur (100%) Die Modulnote ergibt sich aus der Klausurnote.
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 90 h Eigenstudium: 135 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<p>Empfohlene Bücher zur Begleitung und Vertiefung:</p> <ul style="list-style-type: none"> Teich J., Haubelt C.: "Digitale Hardware/Software-Systeme: Synthese und Optimierung", Springer-Verlag, 2007, ISBN: 978-3-540-46822-6 Gajski, D. et al.: "Specification and Design of Embedded Systems", Prentice Hall, Englewood Cliffs, NJ, 1994, ISBN: 978-0131507319 <p>Weitere Informationen:</p> <p>https://www.cs12.tf.fau.de/lehre/lehveranstaltungen/vorlesungen/hardware-software-co-design</p>

1	Modulbezeichnung 93201	Theoretische Informatik für Wirtschaftsinformatik und Lehramt Theoretical computer science for information systems and teaching degree students	5 ECTS
2	Lehrveranstaltungen	Übung: ÜThInfWI-A (2 SWS) Vorlesung: Theoretische Informatik für Wirtschaftsinformatik und Lehramt (2 SWS) Übung: Intensivübung zu Theoretische Informatik für Wirtschaftsinformatik und Lehramt (optional) (2 SWS)	- - -
3	Lehrende	Simon Prucker apl. Prof. Dr. Stefan Milius	

4	Modulverantwortliche/r	apl. Prof. Dr. Stefan Milius	
5	Inhalt	<p>Grundlegende Begriffe und Kernergebnisse der Automatentheorie, Berechenbarkeitstheorie und Komplexitätstheorie werden überblickhaft behandelt:</p> <ul style="list-style-type: none"> • endliche Automaten und reguläre Grammatiken und Sprachen • Kellerautomaten, kontextfreie Grammatiken und Sprachen • Turingmaschinen und berechenbare Funktionen • Primitiv rekursive und mü-rekursive Funktionen • LOOP- und WHILE-Berechenbarkeit • Entscheidbare Sprachen und Unentscheidbarkeit • Chomsky-Hierarchie • Komplexitätsklassen P und NP • NP-Vollständigkeit 	
6	Lernziele und Kompetenzen	<p>Fachkompetenz Wissen Die Studierenden geben elementare Definitionen und Fakten zu formalen Sprachen und entsprechenden Maschinenmodellen und Grammatiken wieder.</p> <p>Verstehen Die Studierenden</p> <ul style="list-style-type: none"> • erklären grundlegende Konzepte der Begriffe der Automaten- und Berechenbarkeits- und Komplexitätstheorie. • beschreiben Beispiele dieser Konzepte. • erläutern grundlegende Konstruktionen, Algorithmen und wesentliche Resultate und entsprechende Beweise (z.B. Unentscheidbarkeit des Halteproblems). <p>Anwenden Die Studierenden</p> <ul style="list-style-type: none"> • führen Konstruktionen auf vorgelegten Maschinen und Grammatiken und Sprachen durch (z.B. Automatenminimierung, Potenzmengen-Konstruktion, Chomsky-Normierung, CYK-Algorithmus). • wenden grundlegende Beweisverfahren der theoretischen Informatik an (z.B. Induktionsbeweise, Pumping-Lemma, Reduktionen). <p>Analysieren</p>	

		<p>Die Studierenden</p> <ul style="list-style-type: none"> • analysieren formale Sprachen und ermitteln ihre Zugehörigkeit zu den Klassen der Chomsky-Hierarchie. • untersuchen die Entscheidbarkeit von vorgelegten formalen Sprachen. • analysieren die Komplexität eines Entscheidungsproblems und klassifizieren es als Problem in P, NP bzw. NP-vollständig. <p>Lern- bzw. Methodenkompetenz Die Studierenden</p> <ul style="list-style-type: none"> • beherrschen das grundsätzliche Konzept des Beweises als hauptsächliche Methode des Erkenntnisgewinns in der theoretischen Informatik. Sie überblicken abstrakte Begriffsarchitekturen. • vollziehen mathematische Argumentationen nach, erklären diese, führen diese selbst und legen sie schriftlich nieder. <p>Sozialkompetenz Die Studierenden lösen Probleme in kollaborativer Gruppenarbeit und präsentieren erarbeitete Lösungen.</p>
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 2
9	Verwendbarkeit des Moduls	Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Klausur (90 Minuten)
11	Berechnung der Modulnote	Klausur (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 56 h Eigenstudium: 94 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> • U. Schöning: Theoretische Informatik - kurz gefasst, 5. Aufl., Spektrum 2008. • J.E. Hopcroft, R. Motwani und J.D. Ullman: Introduction to Automata Theory, Languages, and Computation, 2. Aufl., Addison Wesley, 2001.

Praktikum CE

1	Modulbezeichnung 93116	AI-1 Systems Project AI-1 Systems project	10 ECTS
2	Lehrveranstaltungen	Projektseminar: AI-1 Systems Project (4 SWS)	10 ECTS
3	Lehrende	Jan Frederik Schaefer Prof. Dr. Michael Kohlhase	

4	Modulverantwortliche/r	Prof. Dr. Michael Kohlhase	
5	Inhalt	<p>This project complements the symbolic AI methods covered in the AI-1 Lecture with practical experience in implementing the algorithms or applying existing state-of-the-art libraries and systems. Practical areas of covered in this project include</p> <ul style="list-style-type: none"> • Logic Programming • Rational Agents • Search (classical/adversarial) • Constraint Satisfaction/Propagation • Propositional/First-Order Reasoning • Knowledge Representation • Planning <p>Project participants will work on multiple concrete problems/applications in small project teams. The results will be</p> <ul style="list-style-type: none"> • evaluated using benchmark/problems and/or competitions between project teams. • documented in (short) project reports and be presented to the group. 	
6	Lernziele und Kompetenzen	<p>Students will</p> <ul style="list-style-type: none"> • develop and implement algorithms or apply existing systems/libraries to multiple symbolic AI Problems, • acquire hands-on experience in an established research field, • learn best practices in software development and documentation, • gain first experience in academic writing. 	
7	Voraussetzungen für die Teilnahme	Strong programming skills in any programming language.	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	Praktikum CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	<p>Praktikumsleistung</p> <p>The project consists of several large tasks (≈6). The first task is solved individually, the other tasks are solved in teams of 2 students. A presentation is given on one of the tasks (typically as a group presentation where each student gets a ≈10 minute segment that is individually graded). A written report (no restrictions on page number, 10–15 pages is typical) is submitted on one task. Three reviews for other students' reports have to be written. The reviewed students can use the reviews to improve their report. The review process is double-blind and students can choose not to get reviews for their report.</p>	

11	Berechnung der Modulnote	Praktikumsleistung (100%) The final grade is composed from the quality of the deliverables, namely the practical part (65% of the grade), the presentation (10% of the grade), the written report and reviews (25% of the grade).
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 40 h Eigenstudium: 260 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	

1	Modulbezeichnung 93192	Hackerpraktikum Bachelor Hacking lab (Bachelor's students)	10 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Felix Freiling	
5	Inhalt	<p>In diesem Praktikum lernen die Teilnehmer den kritischen Umgang mit offensiver IT-Sicherheit. Es werden prinzipielle Angriffskonzepte erörtert und in einer abgeschotteten Umgebung gezielt zur Anwendung gebracht. Durch diese praktischen Erfahrungen aus der Sichtweise eines "Hackers" werden die Teilnehmer bzgl. sicherheitsrelevanten Fragen sensibilisiert und können die gewonnenen Erkenntnisse dann letztendlich auch zur Absicherung von Systemen einsetzen. Das Hackerpraktikum wird in bis zu 6 Übungsblätter zu je 3 Wochen aufgeteilt, wobei die folgenden beispielhaften Themen bearbeitet werden:</p> <ul style="list-style-type: none"> • Netzwerksicherheit (Sniffing, Spoofing, WPA, ...) • Webhacking (SQL Injections, XSS, CSRF, ...) • Systemsicherheit (Rootkits, Privileges, Suid, ...) • Reverse Engineering (Cracking, Malware Analysis, ...) • Exploitation (Buffer Overflows, Shellcode, ASLR, ...) <p>Von den Übungen müssen insgesamt 2/3 und pro Blatt 1/2 der möglichen Punkte erreicht werden. Die Übungen sind in Einzelarbeit abzugeben.</p>	
6	Lernziele und Kompetenzen	Die Studierenden können Schwachstellen in den vorgestellten Themenbereichen identifizieren und beschreiben. Sie können aktuelle Angriffs- und Verteidigungstechniken in konkreten Fällen auswählen und anwenden.	
7	Voraussetzungen für die Teilnahme	<ul style="list-style-type: none"> • Interesse an IT-Sicherheit. • Bereitschaft, viel Zeit in das Praktikum zu investieren. • Programmierkenntnisse in C/Assembler und mindestens einer Skriptsprache (bspw. Python). • Linux-Kenntnisse. • Von großem Vorteil sind darüber hinaus Netzwerk-Kenntnisse und Vorkenntnisse im Bereich IT-Sicherheit. 	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	Praktikum CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Praktikumsleistung	
11	Berechnung der Modulnote	Praktikumsleistung (100%)	

12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 50 h Eigenstudium: 250 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	

1	Modulbezeichnung 182798	Supercomputing Praktikum Student cluster competition	10 ECTS
2	Lehrveranstaltungen	Praktikum: Supercomputing Praktikum (8 SWS)	10 ECTS
3	Lehrende		

4	Modulverantwortliche/r	Prof. Dr.-Ing. Dietmar Fey	
5	Inhalt	<p>Das Supercomputer Praktikum bereitet die Studierenden auf eine Teilnahme an der Student Cluster Competition(SCC). Im Rahmen des Wettbewerbs muss ein Teil eines Supercomputers so eingerichtet werden, dass klassische Benchmarks und andere HPC-Anwendungen möglichst performant und effizient ausgeführt werden können. Die Studierenden bearbeiten dafür im Semester die folgenden Themen:</p> <ul style="list-style-type: none"> - Aufbau und Komponenten von Supercomputern - Benchmarking von Anwendung bezüglich Rechenleistung und Energieverbrauch in HPC-Systemen - Konfiguration, Administration und Debugging von Supercomputern <p>Des Weiteren lernen die Studierenden sich selbstständig als Team zu organisieren und den Ablauf und Teilnahme eines Bewerbungsprozesses in einem professionellen Umfeld.</p>	
6	Lernziele und Kompetenzen	<p>Studierende...</p> <ul style="list-style-type: none"> - haben alle notwendigen Fähigkeiten erworben, um an dem Student Cluster Competition auf der SC oder ISC teilzunehmen, - können einen Cluster-Computer ingenieurmäßig entwerfen und zusammenbauen, - verstehen und konfigurieren die speziellen Hardware-Architekturen von HPC-Systemen, - können HPC-Anwendungen installieren, messen und optimieren, - können den Zustand des Systems zu überwachen und es bei Bedarf zu reparieren. 	
7	Voraussetzungen für die Teilnahme	<ul style="list-style-type: none"> - Grundkenntnisse der UNIX-Systeme und der Programmierung in C oder C++ - Bereitschaft in die USA zu reisen um am Wettbewerb teilzunehmen (Reisekosten werden übernommen) 	
8	Einpassung in Studienverlaufsplan	Semester: 6	
9	Verwendbarkeit des Moduls	Praktikum CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Praktikumsleistung	
11	Berechnung der Modulnote	Praktikumsleistung (100%)	
12	Turnus des Angebots	nur im Sommersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 100 h Eigenstudium: 200 h	

14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	

1	Modulbezeichnung 240715	Grafik-Praktikum Game Programming Laboratory course: Games programming (GraPra)	10 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Dr.-Ing. Frank Bauer Prof. Dr. Tim Weyrich	
5	Inhalt	Wie entstehen eigentlich digitale Inhalte in einem Museum? In diesem Projekt-Seminar lernen Sie den zugrundeliegenden Ablauf in Zusammenarbeit mit dem Germanischen Nationalmuseum kennen. Dabei arbeiten Sie selbst an jedem Schritt mit: vom Scan im Museum über die 3D-Rekonstruktion und Aufbereitung der Daten (in Blender) bis zur inhaltlichen Gestaltung (mit Unity 3D) und finalen Veröffentlichung z.B. über Sketchfab. Durch das Semester werden Sie immer begleitet von Experten aus dem Germanischen Nationalmuseum, Mitarbeitern der Computergrafik und der Digital Humanities in Erlangen. So können Sie im Laufe des Semesters eine digitale Ausstellung von Anfang bis Ende interaktiv und spannend gestalten.	
6	Lernziele und Kompetenzen	Fachkompetenz Erschaffen Lernende planen, entwerfen und produzieren unter Zuhilfenahme von typischen Werkzeugen der Computergrafik eine digitale Ausstellung für eine reales Museumsobjekt.	
7	Voraussetzungen für die Teilnahme	Sie sollten bereits über grundlegende Programmierkenntnisse verfügen. Mögliche Vorlesungen sind z.B. AuD, IWGS oder Gdl. Idealerweise haben Sie auch bereits Erfahrung im Umgang mit 3D-Software oder die Veranstaltung Computergrafik absolviert.	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	Praktikum CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Praktikumsleistung	
11	Berechnung der Modulnote	Praktikumsleistung (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 40 h Eigenstudium: 260 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Deutsch	
16	Literaturhinweise		

1	Modulbezeichnung 547324	Praktikum: Entwicklung interaktiver eingebetteter Systeme (unbenotet) Laboratory course: Development of interactive embedded systems (ungraded)	10 ECTS
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	Modulverantwortliche/r	
5	Inhalt	keine Inhaltsbeschreibung hinterlegt!
6	Lernziele und Kompetenzen	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Praktikum CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Praktikumsleistung
11	Berechnung der Modulnote	Praktikumsleistung (bestanden/nicht bestanden)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	Dauer des Moduls	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 1995	Industriepraktikum (B.Sc. Computational Engineering 2010) Internship / practical training on industry	10 ECTS
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	Modulverantwortliche/r	
5	Inhalt	keine Inhaltsbeschreibung hinterlegt!
6	Lernziele und Kompetenzen	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	keine Einpassung in Studienverlaufsplan hinterlegt!
9	Verwendbarkeit des Moduls	Praktikum CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Praktikumsleistung (8 Wochen)
11	Berechnung der Modulnote	Praktikumsleistung (bestanden/nicht bestanden)
12	Turnus des Angebots	keine Angaben zum Turnus des Angebots hinterlegt!
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	Dauer des Moduls	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

Seminar Informatik für CE

1	Modulbezeichnung 47637	Geschichte der Rechentechnik History of computing	5 ECTS
2	Lehrveranstaltungen	Hauptseminar: Die Geschichte der Rechentechnik I (2 SWS) Übung: Übung zu Geschichte der Rechentechnik (2 SWS)	5 ECTS -
3	Lehrende	Dr. phil. Felix Schmutterer	

4	Modulverantwortliche/r	Dr. phil. Felix Schmutterer	
5	Inhalt	<p>Gegenstand des Seminars sind die Meilensteine der Rechentechnik" ausgehend vom 19. Jahrhundert. Diese einschneidenden Entwicklungen von Rechenmaschinen zu ersten Werkzeugen der Datenverarbeitung werden zunächst den Ausgangspunkt bilden. Turingmaschinen" und die neuen Bedürfnisse" von Militär wie etwa Chiffrierung und De-Chiffrierung werden dann zentrale Themen des Seminars bilden. Im Fokus steht dabei stets die Funktionsweise der Maschinen. Darüber hinaus werden die Rechner konsequent im Kontext ihrer Zeit diskutiert werden. Insbesondere wird dabei auf die steigenden Anforderungen und die veränderlichen Einsatzmöglichkeiten wie etwa im Falle der Enigma einzugehen sein.</p> <p>Die genauen Themen werden zu Semesterbeginn festgelegt. Die Themenliste kann beim Dozenten erfragt werden.</p>	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> recherchieren und arbeiten mit historischen Quellen und wissenschaftlicher Literatur aus den Bereichen Informatik und Geschichte beschreiben Aspekte der Rechentechnik erarbeiten sich die Fähigkeit, wichtige Aspekte für einen wissenschaftlichen Vortrag darzustellen und strukturieren diesen vertreten ihre Auffassung in einer Diskussion und hinterfragen ihr Thema konzipieren und formulieren eine schriftliche Zusammenfassung des Vortrags 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 3	
9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Seminarleistung (30 Minuten) Referat und schriftliche Hausarbeit (10 Seiten)	
11	Berechnung der Modulnote	Seminarleistung (100%) Die Note ist gewichtet, das Referat fließt zu 40% ein, die Hausarbeit zu 60%	
12	Turnus des Angebots	in jedem Semester	

13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 93127	Maschinelles Lernen: Einführung Machine learning: Introduction	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen. Anwesenheit bei den Vorträgen der anderen Teilnehmer erwünscht .	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Michael Philippsen	
5	Inhalt	<p>Dieses Seminar führt in das Themengebiet des maschinellen Lernens (ML) ein. ML ist die Wissenschaft, Computer zum Handeln zu bewegen, ohne explizit programmiert zu werden. ML ist heute so allgegenwärtig, dass wir es wahrscheinlich täglich verwenden, ohne es zu wissen. So hat ML in den letzten Jahren beispielsweise selbstfahrende Autos, praktische Bild- und Spracherkennung und die effektive Partner- und Websuche ermöglicht.</p> <p>Ziel des Seminars ist eine umfassende Einführung in das maschinelle Lernen, Analyse und Verarbeitung von Daten sowie statistische Mustererkennung. Zu den Themen gehören: (1) Klassifizierungs- und Regressionsprobleme; (2) überwachtes Lernen (parametrische und nicht parametrische Algorithmen, lineare und logistische Regression, k-nächster Nachbar, Support-Vector-Machines, Entscheidungsbäume, flache neuronale Netze); (3) unüberwachtes Lernen (K-Means, Clustering, Dimensionsreduktion, PCA, LDA, Empfehlungssysteme); (4) Ensemble- und Online-Lernen; (5) Regularisierung: Modelldiagnose, Fehleranalyse und Qualitätsmetriken sowie Interpretation der Ergebnisse; (5) evolutionäre Algorithmen; (6) Anomalieerkennung und Gaußsche Verteilungen; (7) Bayes, Kalman-Filter und Gaußsche Prozesse. Die genannten Themen sind an den aktuellen Forschungsstand angepasst und wechseln sich jährlich ab. Das Seminar gibt einen Einblick in die Welt des maschinellen Lernens und befähigt den Studierenden eine wissenschaftliche Präsentation und Ausarbeitung anzufertigen, um individuell erworbenes Wissen einem Fachpublikum vermitteln zu können.</p>	
6	Lernziele und Kompetenzen	<p>Die Teilnahme an diesem Seminar ermöglicht den Studierenden sich in der Kompetenz maschinelles Lernen auszubilden und erlerntes Wissen in Form einer angeleiteten Präsentation und Ausarbeitung wissenschaftlich darzustellen und zu kommunizieren: Die Studierenden erlangen durch das Seminar die Kompetenz und das Wissen:</p> <ul style="list-style-type: none"> • prinzipielle Vorgehensweisen beim maschinellen Lernen zu erläutern, • Vor- und Nachteile einzelner Methoden zu untersuchen, • Chancen und Grenzen des maschinellen Lernens zu erläutern, • Sachverhalte unter Fachleuten zu diskutieren, • fachspezifische Fragen für das Gebiet zu beantworten, 	

		<ul style="list-style-type: none"> • Konzepte des Maschinellen Lernens im Allgemeinen und deren Anwendung in Applikationsgebieten der Industrie, Sozialwesen, Bildung und Sport zu erlernen, • Datenvorverarbeitung, ML-Methoden und Interpretation der Ergebnisse in konkreten Fragestellungen zu modellieren und zu adaptieren. <p>Weiter führt das Seminar die Studierende in das wissenschaftliche Arbeiten ein, um selbstständig:</p> <ul style="list-style-type: none"> • erforderliche Literatur aufzufinden, zu analysieren und zu bewerten, • sich eigenständig in ein Themengebiet einzuarbeiten, • Grundzüge der Präsentationstechniken anzuwenden und zu motivieren, • eine Präsentation mit Begleitmaterial für ein Fachpublikum zu entwickeln, • einen Vortrag passend für einen vorgegebenen Zeitrahmen durchzuführen, • eine Ausarbeitung im Stil einer wissenschaftlichen Publikation mit Latex anzufertigen, • Sprache, Sprachangemessenheit, Inhalt sowie Aufbau und die wissenschaftliche Darstellung einer wissenschaftlichen Ausarbeitung zu verinnerlichen, • und die eigene Kognition und Kreativität in der Ausarbeitung zu bewerten.
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>Seminarleistung</p> <p>Die Gesamtnote setzt sich zu 50% aus der Bewertung des Vortrags und zu 50% aus der Bewertung der Ausarbeitung / Implementierung zusammen:</p> <ul style="list-style-type: none"> • 45-60 Minuten Seminarvortrag. • Erstellung einer Ausarbeitung mit den wesentlichen Punkten des Vortrags (keine Folienkopien, ca. 6-8 Seiten im IEEE Format für Konferenzbeiträge). • Alternativ zur Ausarbeitung kann eine Demonstration implementiert werden. In diesem Fall umfasst die Ausarbeitung (Dokumentation der Implementierung) lediglich ca. 3-4 Seiten im IEEE Format für Konferenzbeiträge. • Vortrag und Ausarbeitung sollten auf Englisch erfolgen. • Fertigstellung der Folien bis spätestens einer Woche vor dem Vortragstermin. • Fertigstellung der Ausarbeitung bis zum Ende des Semesters.
11	Berechnung der Modulnote	Seminarleistung (100%)

		Die Gesamtnote setzt sich zu 50% aus der Bewertung des Vortrags und zu 50% aus der Bewertung der Ausarbeitung / Implementierung zusammen.
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 30 h Eigenstudium: 120 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Englisch
16	Literaturhinweise	<ul style="list-style-type: none"> • A. Müller und S. Guido: Introduction to Machine Learning with Python: A Guide for Data Scientists, O'Reilly UK Ltd., 2016 • K. P. Murphy: Machine learning - a probabilistic perspective, Adaptive computation and machine learning series, MIT Press, 2012. • T. J. Hastie und R. Tibshirani und J. H. Friedman: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer Series in Statistics, 2009. • T. M. Mitchell: Machine Learning, McGraw-Hill Education Ltd., 1997 • F. V. Jensen: An Introduction To Bayesian Networks, Springer, 1996 • J. A. Freeman: Simulating neural networks - with Mathematica, Addison-Wesley Professional, 1993 • J. A. Hertz und A. Krogh und R. G. Palmer: Introduction to the theory of neural computation, Westview Press, 1991 • R. Rojas: Theorie der neuronalen Netze - eine systematische Einführung, Springer, 1993 • W. Banzhaf und F. D. Francone und R. E. Keller und P. Nordin: Genetic programming - An Introduction: On the Automatic Evolution of Computer Programs and Its Applications, Morgan Kaufmann, 1998 • M. Mitchell: An introduction to genetic algorithms, MIT Press, 1996 • Z. Michalewicz: Genetic Algorithms + Data Structures = Evolution Programs, Springer, 1992 • M. Bishop: Pattern Recognition and Machine Learning (Information Science and Statistics), Springer, 2006

1	Modulbezeichnung 93656	Seminar Energieinformatik Seminar: Energy informatics	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Marco Pruckner	
5	Inhalt	Teilnehmende arbeiten sich selbständig anhand der ausgewählten wissenschaftlichen Literatur in ein vorgeschlagenes Thema aus dem Bereich der Energieinformatik ein, erstellen dazu eine schriftliche Ausarbeitung, bereiten einen Seminarvortrag vor und präsentieren ihn vor Lehrenden, interessierten wissenschaftlichen Mitarbeitern des Instituts und Kommilitonen der Lehrveranstaltung.	
6	Lernziele und Kompetenzen	<p>Zu den zu erwerbenden Kompetenzen zählen</p> <ul style="list-style-type: none"> • die Literaturrecherche, • korrektes Zitieren, • die Fähigkeit auszuwählen, welche Aspekte in der Ausarbeitung und im Vortrag behandelt werden, • zielgruppengerechtes Schreiben, • die Verwendung von Textverarbeitungswerkzeugen für die Präsentation und die Ausarbeitung, • sicheres Auftreten beim Vortragen eines wissenschaftlichen Themas sowie • fachspezifische Fragen zum ausgewählten Themengebiet zu beantworten. <p>Am Ende jeder Lehreinheit</p> <ul style="list-style-type: none"> • stellen die Teilnehmenden gezielt Fragen zum vorgetragenen Thema, • diskutieren über fachliche Aspekte der Präsentation, • üben konstruktive Kritik an der Darstellung des Themas und • bewerten die Präsentation aus vortragstechnischer Sicht. 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 5	
9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Seminarleistung	
11	Berechnung der Modulnote	Seminarleistung (100%)	
12	Turnus des Angebots	nur im Wintersemester	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 30 h Eigenstudium: 120 h	

14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none">• gesammelte Bücher, Artikel und Weblinks zu einem einschlägigen Themengebiet• Marco Pruckner. Allgemeine Hinweise zu Seminarvorträgen und -ausarbeitungen.

1	Modulbezeichnung 93096	Iterative Lösungsverfahren für lineare und nichtlineare Gleichungssysteme Iterative solution methods for linear and non-linear systems	5 ECTS
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	Modulverantwortliche/r	Prof. Dr. Christoph Pflaum	
5	Inhalt	<ul style="list-style-type: none"> • Einführung in die einzelnen Phasen der Softwareentwicklung: Anforderungsanalyse, Spezifikation, Entwurf, Implementierung, Test, Wartung • Beispielhafter Einsatz ausgewählter repräsentativer Verfahren zur Unterstützung dieser Entwicklungsphasen • Ergonomische Prinzipien Benutzungsoberfläche • Objektorientierte Analyse und Design mittels UML • Entwurfsmuster als konstruktive, wiederverwendbare Lösungsansätze für ganze Problemklassen • Automatisch unterstützte Implementierung aus UML-Diagrammen • Teststrategien • Refactoring zur Unterstützung der Wartungsphase 	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • wenden auf Basis der bereits erworbenen Programmierkenntnisse systematische und strukturierte Vorgehensweisen (wie das Wasserfall- und V-Modell) zur Bewältigung der Komplexität im Zusammenhang mit dem "Programmieren-im-Großen" an; • benutzen ausgewählte Spezifikationssprachen (wie Endliche Automaten, Petri-Netze und OCL), um komplexe Problemstellungen eindeutig zu formulieren und durch ausgewählte Entwurfsverfahren umzusetzen; • wenden UML-Diagramme (wie Use Case-, Klassen-, Sequenz- und Kommunikationsdiagramme) zum Zweck objektorientierter Analyse- und Design-Aktivitäten an; • reproduzieren allgemeine Entwurfslösungen wiederkehrender Probleme des Software Engineering durch Verwendung von Entwurfsmustern; • erfassen funktionale und strukturelle Testansätze; • setzen Refactoring-Strategien zur gezielten Erhöhung der Software-Änderungsfreundlichkeit um. 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 3	

9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Seminarleistung
11	Berechnung der Modulnote	Seminarleistung (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	
16	Literaturhinweise	Lehrbuch der Softwaretechnik (Band 1), Helmut Balzert, 2000

1	Modulbezeichnung 97008	Advanced Design and Programming (5-ECTS) Advanced design and programming (5-ECTS)	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Dirk Riehle	
5	Inhalt	<p>This course teaches principles and practices of advanced object-oriented design and programming. Dieser Kurs wird auf Deutsch gehalten. It consists of a weekly lecture with exercises, homework and self-study. This is a hands-on course and students should be familiar with their Java IDE. Students learn the following concepts:</p> <p>Class-Level</p> <ul style="list-style-type: none"> • Method design • Class design • Classes and interfaces • Subtyping and inheritance • Implementing inheritance • Design by contract <p>Collaboration-Level</p> <ul style="list-style-type: none"> • Values vs. objects • Role objects • Type objects • Object creation • Collaboration-based design • Design patterns <p>Component-Level</p> <ul style="list-style-type: none"> • Error handling • Meta-object protocols • Frameworks and components • Domain-driven design • API evolution <p>The running example is the photo sharing and rating software Wahlzeit, see https://github.com/dirkriehle/wahlzeit . Class is held as a three hour session with a short break in between. Students should have a laptop ready with a working Java programming setup. Sign-up and further course information are available at https://adap.uni1.de - please sign up for the course on StudOn (available through previous link) as soon as possible. The course information will also tell you how the course will be held (online or in person).</p>	
6	Lernziele und Kompetenzen	<ul style="list-style-type: none"> • Students learn to recognize, analyze, and apply advanced concepts of object-oriented design and programming 	

		<ul style="list-style-type: none"> Students learn to work effectively with a realistic tool set-up, involving an IDE, configuration management, and a service hoster
7	Voraussetzungen für die Teilnahme	INF-AuD or compatible / equivalent course
8	Einpassung in Studienverlaufsplan	Semester: 3
9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010 Technische Wahlmodule Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Variabel The specifics of the examination is aligned with the didactic character of the module and will be clarified by the examiner no later than two weeks after the start of the lecture. Die Konkretisierung der Prüfungsform und -umgang richtet sich nach dem didaktischen Charakter des Moduls und wird bis spätestens zwei Wochen nach Vorlesungsbeginn durch den Prüfer bekannt gegeben.
11	Berechnung der Modulnote	Variabel (100%)
12	Turnus des Angebots	nur im Wintersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	<ul style="list-style-type: none"> See https://adap.uni1.de

1	Modulbezeichnung 349413	Seminar Sprachtechnologie für Sprachpathologien Speech technologies for speech pathologies	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Seung Hee Yang	
5	Inhalt	This seminar deals with how the diagnosis and therapy of different speech pathologies can be supported by speech technology. The participants should present selected speech, speech and voice disorders in a lecture and demonstrate corresponding technologies in the field of pattern recognition and speech processing.	
6	Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> • erlernen die Literaturrecherche. • arbeiten sich selbstständig anhand der gefundenen Literatur in die Thematik der automatischen Analyse von Sprach-, Sprech- und Stimmstörungen ein. • wählen einen Schwerpunkt und bereiten diesen im Rahmen einer Präsentation so auf, dass er für andere Teilnehmer des Seminars verständlich ist. • lernen die Anforderungen an einen wissenschaftlichen Vortrag auf einer internationalen Konferenz kennen. • halten einen Vortrag in der international üblichen Fachsprache Englisch (davon ausgenommen sind Studierende aus dem Ausland, die in Deutschland studieren, um Deutsch zu lernen) 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Seminarleistung Die Seminarleistung setzt sich zusammen aus der Bewertung einer schriftlichen Ausarbeitung in Form der Folien des Vortrags (30% der Note) und einem Vortrag (70% der Note)	
11	Berechnung der Modulnote	Seminarleistung (100%)	
12	Turnus des Angebots	Unregelmäßig	
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 30 h Eigenstudium: 120 h	
14	Dauer des Moduls	1 Semester	
15	Unterrichts- und Prüfungssprache	Englisch	

1	Modulbezeichnung 628256	Seminar Machine Learning Algorithms Seminar machine learning algorithms	5 ECTS
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	Modulverantwortliche/r	
5	Inhalt	keine Inhaltsbeschreibung hinterlegt!
6	Lernziele und Kompetenzen	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 5
9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Seminarleistung
11	Berechnung der Modulnote	Seminarleistung (100%)
12	Turnus des Angebots	in jedem Semester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	Dauer des Moduls	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 921878	Blender Seminar Blender seminar	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Marc Stamminger	
5	Inhalt	In diesem Modul vermitteln wir grundlegende Kenntnisse über den Umgang mit 3D-Modellierungstools und zur Planung von Projekten anhand der OpenSource Software BlendER.	
6	Lernziele und Kompetenzen	<p>Fachkompetenz Wissen Studierende können...</p> <ul style="list-style-type: none"> • ... mathematische Grundlagen verschiedener Beleuchtungsmodelle darlegen • ... Konzepte der Szenengestaltung und Beleuchtung reproduzieren • ... Mathematische Grundlagen zu Interpolationsverfahren darlegen • ... sich an Lizenzmodellen für eigene Werke erinnern • ... über verschiedene Strategien zur Projekt- und Teamplanung berichten <p>Verstehen Studierende können...</p> <ul style="list-style-type: none"> • ... verschiedene Verfahren zur Keyframe- oder Vertexinterpolation schildern • ... Probleme bei der Erstellung von Geometrie aufzeigen • ... Probleme des Photon-Tracing illustrieren <p>Anwenden Studierende können...</p> <ul style="list-style-type: none"> • ... Wissen über Szenengestaltung und Beleuchtung auf konkrete Beispiele anwenden • ... eigene Materialshader auf der Grundlage vorgegebener Formeln/Vorschriften in BlendER umsetzen <p>Analysieren</p> <ul style="list-style-type: none"> • Selbstständige Projektplanung, Teamorganisation und Umsetzung eines 3D-Projektes in vorgegebenem Zeitfenstern 	
7	Voraussetzungen für die Teilnahme	Keine	
8	Einpassung in Studienverlaufsplan	Semester: 4	
9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010	
10	Studien- und Prüfungsleistungen	Seminarleistung	

11	Berechnung der Modulnote	Seminarleistung (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 80 h Eigenstudium: 70 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch Englisch
16	Literaturhinweise	

1	Modulbezeichnung 941318	Neuartige Rechnerarchitekturen Innovative computer architectures	5 ECTS
2	Lehrveranstaltungen	Hauptseminar: Neuartige Rechnerarchitekturen (4 SWS)	5 ECTS
3	Lehrende	Philipp Holzinger Thomas Schlögl Prof. Dr.-Ing. Dietmar Fey John Reuben Prabahar	

4	Modulverantwortliche/r	Prof. Dr.-Ing. Dietmar Fey	
5	Inhalt	<p>Die Entwicklung moderner CPUs hat eine interessante Evolution durchlaufen. Angefangen bei einfachen Single-Core CPUs wurde zunächst die Taktschraube immer weiter nach oben gedreht. Als dies aus thermischen Grund nicht weiter möglich war, wurden Parallelrechner aus ihrer akademischen Nische vertrieben und zum Allgemeingut eines jeden Informatikers. Neuere Entwicklung zeigen nun den Einsatz von heterogenen Rechnerarchitekturen, also die Verbindung verschiedener Recheneinheiten wie CPUs, GPUs, FPGAs, um mittels Spezialhardware anfallende Aufgaben schneller und energieeffizienter lösen zu können. Neuste Forschungsansätze hingegen versuchen nun auch den Hauptspeicher eines Rechners "intelligent" zu machen und Prozessoren direkt in den Speicher zu integrieren - sogenanntes in- oder near-memory-Computing.</p> <p>Ziel dieses Moduls ist das ...</p> <ul style="list-style-type: none"> ... kennen, ... verstehen, ... verwenden, ... vergleichen, und evaluieren <p>verschiedener Rechnerarchitekturen von der Multi-Core CPU bis zum FPGA-Near-Memory-Beschleuniger. Anhand praktischer Anwendungen (z.B. Neuronale Netze, Bildverarbeitung, Autonomes Fahren) können die Architekturen erprobt werden.</p> <p>Hierzu wird jedem Teilnehmenden ein Thema/Architektur zur Bearbeitung übertragen, welche sie/er selbstständig wissenschaftlich in einer schriftlichen Ausarbeitung und didaktisch in einem Vortrag aufarbeitet und präsentiert.</p>	
6	Lernziele und Kompetenzen	<p>Fachkompetenz</p> <p>Wissen</p> <p>Lernende können Wissen über die Grundprinzipien moderner Rechnerarchitekturen (Intel, ARM CPUs; AMD, Nvidia GPUs; FPGAs, Beschleunigerkerne) wiedergeben.</p> <p>Verstehen</p> <p>Lernende verstehen die Grundprinzipien der Datenverarbeitung der einzelnen Architekturen; im speziell verstehen sie ob und warum eine vorgegebene Architektur besonders gut für die Lösung eines Problems geeignet ist.</p>	

		<p>Lernende verstehen die unterschiedlichen Ansätze zur Parallelismus der vorgestellten Architekturen.</p> <p>Anwenden</p> <p>Lernende sind in der Lage Anwendungen auf den vorgegebenen Architekturen z.B. durch Programmierung umzusetzen. Hierzu erklären Studierende wie die Parallelisierungstechniken in bestehenden Architekturen eingesetzt werden.</p> <p>Evaluieren (Beurteilen)</p> <p>Lernende evaluieren die Eignung von Architekturen, bestimmte Probleme effizient auf diese Abbilden zu können.</p> <p>Sozialkompetenz</p> <p>Lernende können komplexe fachbezogene Inhalte klar und zielgruppengerecht präsentieren und eigene Standpunkte in einer Fachdiskussion argumentativ vertreten.</p>
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	<p>Seminarleistung</p> <p>Die Seminarleistung setzt sich wie folgt zusammen: ca. 10 Seiten Ausarbeitung und 25 Minuten Präsentation mit 50:50 Gewichtung bei der Notenfindung.</p>
11	Berechnung der Modulnote	Seminarleistung (100%)
12	Turnus des Angebots	nur im Sommersemester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 60 h Eigenstudium: 90 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	

1	Modulbezeichnung 949119	Algorithmen der Simulationstechnik Algorithms of simulation technology	5 ECTS
2	Lehrveranstaltungen	Im aktuellen Semester werden keine Lehrveranstaltungen zu dem Modul angeboten. Für weitere Auskünfte zum Lehrveranstaltungsangebot kontaktieren Sie bitte die Modul-Verantwortlichen.	
3	Lehrende	-	

4	Modulverantwortliche/r	Prof. Dr. Ulrich Rüde
5	Inhalt	keine Inhaltsbeschreibung hinterlegt!
6	Lernziele und Kompetenzen	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	Voraussetzungen für die Teilnahme	Keine
8	Einpassung in Studienverlaufsplan	Semester: 4
9	Verwendbarkeit des Moduls	Seminar Informatik für CE Bachelor of Science Computational Engineering (Rechnergestütztes Ingenieurwesen) 2010
10	Studien- und Prüfungsleistungen	Seminarleistung
11	Berechnung der Modulnote	Seminarleistung (100%)
12	Turnus des Angebots	in jedem Semester
13	Arbeitsaufwand in Zeitstunden	Präsenzzeit: 30 h Eigenstudium: 120 h
14	Dauer des Moduls	1 Semester
15	Unterrichts- und Prüfungssprache	Deutsch
16	Literaturhinweise	