



Fakultät Informatik

Studiengang Informatik (Master)

Modulhandbuch

Stand: WS 2021/22

August 2021

Prof. Dr. E. Müller
Studiendekan der Fakultät Informatik

Prof. Dr. E. Böhrer
Studiengangkoordinator

Prof. Dr. U. Göhner
Vorsitzender der Prüfungskommission

Inhaltsverzeichnis

1 Ziele und Aufbau des Studiengangs Informatik	2
2 Begriffserläuterungen	7
3 Modulbeschreibungen	9
IFM1101 Effiziente Algorithmen	9
IFM1102 Mathematik	11
IFM1103 Softwarearchitektur	13
IFM5201 Data Science	15
IFM5301 Betriebssystem im Eigenbau	17
IFM5302 Echtzeitsysteme	18
IFM5303 Multimodale Sensorsysteme	20
IFM5304 Mobile Roboter	22
IFM5309 Kryptographie	24
IFM5311 Computer Vision	26
IFM5401 Requirements Engineering und Management	28
IFM5402 Identity- und IT-Architekturmanagement	30
IFM5403 eHealth	32
IFM5404 Datenschutz	34
IFM5405 Integration und Management von Geschäftsprozessen mit SAP	35
IFM5406 Funktionale Sicherheit	37
IFM5601 Deep Learning	39
IFM5602 Mustererkennung und maschinelles Lernen	41
IFM5603 Simulation und Datenvisualisierung	43
IFM5604 Parallelprogrammierung	45
IFM5605 Algorithmen und Strategien zur Entscheidungsunterstützung	47
IFM5606 Big Data	49
IFM5607 Algorithmen für Real Time Rendering	51
IFM5607 E-Business Management	53
IFM5100 Hauptseminar	55
IFM5200 Projekt	56
IFM6101 Masterarbeit	57



1 Ziele und Aufbau des Studiengangs Informatik

Ziele des Masterstudiengangs Informatik sind die Vermittlung verschiedenster Kompetenzen und Lehrinhalte. Der Studiengang ist als anwendungsorientierter postgradualer konsekutiver Studiengang konzipiert. Er baut inhaltlich auf den Bachelor- oder Diplomstudiengängen Informatik, Wirtschaftsinformatik und Informatik- Game Engineering auf.

Der Studiengang ist technologieorientiert und deckt zentrale Gebiete der Informatik ab. Er qualifiziert die Studierenden für das Tätigkeitsfeld Informatik sowie für anwendungsorientierte Forschung und legt dabei besonderen Wert auf die theoretisch-wissenschaftlichen Grundlagen.

In der folgenden Tabelle werden den übergeordneten abstrakten Kompetenzen (Studienziele) die jeweiligen Lernergebnisse zugeordnet:

Nr.	Studienziele	Lernergebnisse
1	Formale, algorithmische und mathematische Kompetenz	Fundiertes Wissen über die Grundlagen der Informatik und der Mathematik Fähigkeit das erworbene Wissen in der Praxis anzuwenden und selbständig auszuweiten
2	Analyse-, Design- und Realisierungs-Kompetenz	Fähigkeit zur Lösung von ungewöhnlichen und Nicht-Standardproblemen Fähigkeit aus unvollständigen und widersprüchlichen Informationen in Abstimmung mit dem Auftraggeber konsistente und überzeugende Lösungen zu erarbeiten und umzusetzen
3	Technologische Kompetenz	Vertiefte Kenntnis der technologischen Grundlagen mindestens eines Anwendungsbereichs der Informatik Fähigkeit sich in neue Technologien einzuarbeiten, Technologien auszuwählen und einzusetzen Fähigkeit Technologien wissenschaftlich fundiert weiterzuentwickeln
4	Methodenkompetenz	Fähigkeit das erworbene Wissen auch in komplexen Anwendungsszenarien einzusetzen und erfolgreich anzuwenden Befähigung zu wissenschaftlicher Arbeit, Gute wissenschaftliche Praxis
5	Projektmanagementkompetenz	Fähigkeit auch größere Projekte zu planen, zu organisieren und zu leiten, sowie gegenüber Dritten zu vertreten Probleme frühzeitig zu erkennen und zu lösen und Projekte erfolgreich abzuschließen
6	Anwendungskompetenz	Umfassendes fachliches und technisches Wissen und vertiefte Kenntnis der Vorgehensweisen und Prozesse in einem oder mehreren Anwendungsbereichen der Informatik Verständnis für die rechtlichen und nichttechnischen Auswirkungen von Anwendungen der Informatik

Der Zusammenhang zwischen den übergeordneten Zielen und den Lernergebnissen des Masterstudiengangs sowie der Beitrag der Pflicht- und Wahlpflichtmodule zur Umsetzung dieser Ziele sind in der folgenden Zielematrix dargestellt:

Modul	Studienziel					
	1	2	3	4	5	6
Mathematik	++	+				
Softwarearchitektur		++	+	+		
Effiziente Algorithmen	++		+			
Hauptseminar			+	++		
Projekt	+	+	+	++	++	+
Parallelprogrammierung			++			
Datenschutz				+	+	++
Betriebssystem im Eigenbau			++			
Bussysteme			++			+
Mikrocontroller			++			+
Kryptographie	+	+	++			++
Identity- und IT-Architektur-Management		+		++	++	+
E-Business-Management		+	+	+		++
Requirements Engineering und Management		++		+	+	
Integration und Management von Geschäftsprozessen mit SAP		+	+	+	++	++
Algorithmen und Strategien zur Entscheidungsunterstützung	+		+	++		+
Augmented Reality			+			++
Mustererkennung und maschinelles Lernen	+		+			++
Computer Vision	++		+			++
Multimodale Sensorsysteme			+			++
Echtzeitsysteme			+			++
Deep Learning	+		+			++
Mobile Roboter	+		+			++
eHealth	+	+	+			++
BigData	+	+	+			++
Simulation und Datenvisualisierung	+		++			++
Data Science	+	++		+		+
Masterarbeit	+	++	+	++		++

Absolventen des Masterstudiengangs Informatik erwerben während Ihres Studiums eine Zusatzqualifikation, durch die sie mittels wissenschaftlicher Methodik anspruchsvolle Aufgaben der Informatik bearbeiten können. Während die Absolventen im Bereich der Grundlagen der Informatik und Mathematik ein hervorragendes Fundament erworben haben, befähigt der Masterstudiengang sie das große Potenzial auf Anwendungsseite zu beherrschen. Die Komplexität von Anwendungen steigt unaufhörlich und erfordert eine entsprechende Qualifikation. Die Absolventen sollen aufbauend auf ihren erworbenen fachlichen, methodischen und anwendungsspezifischen Kompetenzen in der Lage sein, komplexe Anwendungsszenarien zu verstehen, problemadäquate Technologien auszuwählen und einzusetzen, zu Problemen auch bei widersprüchlicher und unvollständiger Information Lösungskonzepte zu entwickeln und zu vertreten, Projekte entsprechend zu planen, zu leiten und durchzuführen.

Zielgruppe des Masterstudiengangs sind Absolventen der Informatik, der Wirtschaftsinforma-

tik, Informatik-Game Engineering oder eines gleichwertigen Studiengangs an einer Hochschule.

Das Konzept des Masterstudiengangs in Voll- und Teilzeit beruht auf dem soliden Fundament der drei Pflichtmodule Effiziente Algorithmen, Mathematik und Softwarearchitektur. Auf diesem Fundament bauen die Wahlpflichtmodule auf. Die Wahlpflichtmodule sind den drei Anwendungsbereichen "Autonome Systeme und Technische Informatik", "Wirtschaftsinformatik und E-Business" und "Data Science" zugeordnet. Bei diesen Wahlpflichtmodulen besteht für die Studierenden eine freie Kombinierbarkeit, welche diesen sowohl eine vertiefte Spezialisierung, als auch den Erwerb eines breiten Fachwissens ermöglicht. In den Pflichtveranstaltungen Hauptseminar und Projektarbeit beweisen und vertiefen die Studierenden sowohl anwendungsorientiert-wissenschaftliche als auch methodische- und Teamkompetenz.

Qualifikationsvoraussetzung für den Masterstudiengang in Vollzeit ist ein mindestens mit der Note 2,5 abgeschlossenes Studium der Informatik, der Wirtschaftsinformatik, Informatik-Game Engineering oder eines gleichwertigen Abschlusses. Wurden in einem Bachelorstudiengang weniger als 210 ECTS, mindestens aber 180 ECTS erworben, so sind die bis zum Kompetenzniveau von 300 ECTS fehlenden ECTS-Punkte aus den grundständigen Studiengängen der Hochschule Kempten nachzubringen. Welche Module noch zu erbringen sind bestimmt die Prüfungskommission.

Über die Gleichwertigkeit der Abschlüsse entscheidet die Prüfungskommission. Ist ein Abschluss nicht gleichwertig, kann die Gleichwertigkeit des Abschlusses durch Entscheidung der Prüfungskommission auch dadurch erzielt werden, dass fehlende Kompetenzen im Umfang von max. 30 ECTS-Punkten aus den grundständigen Studiengängen der Hochschule Kempten erworben werden.

Voraussetzung für die Teilnahme am Studiengang in Teilzeit ist das Erfüllen derselben Qualifikationsvoraussetzungen wie im Vollzeitstudiengang.

Aufbau des Studiengangs Informatik (Master)

Studiengang in Vollzeit

Informatik, Master (M.Sc.)

Hochschule für angewandte Wissenschaften Kempten

gültig mit Studienbeginn WS 2017/18

Semester

3	Masterarbeit ⁽⁶⁾																														ECTS-Punkte
2	Projekt ⁽⁴⁾					WP-Fach ⁽⁵⁾					WP-Fach ⁽⁵⁾					WP-Fach ⁽⁵⁾					WP-Fach ⁽⁵⁾										
1	Effiziente Algorithmen ⁽³⁾					Mathematik ⁽¹⁾					Softwarearchitektur ⁽²⁾					Hauptseminar ⁽⁴⁾					WP-Fach ⁽⁵⁾					WP-Fach ⁽⁵⁾					
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	

Legende:

1 Mathematik	4 Angewandte Informatik und interdisziplinäre Fächer
2 Softwaretechnik	5 Wahlpflichtmodul
3 Theorie der Informatik	6 Seminar- und Abschlussarbeit

Zuordnung der Wahlpflichtmodule zu Anwendungsbereichen

	1)	2)	3)
Betriebssystem im Eigenbau			X
Echtzeitsysteme			X
Multimodale Sensorsysteme			X
Deep Learning	X		X
Mobile Roboter			X
Bussysteme			X
Mikrocontroller			X
Mustererkennung und maschinelles Lernen	X		X
Simulation und Datenvisualisierung	X		
Parallelprogrammierung	X		X
Algorithmen und Strategien zur Entscheidungsunterstützung	X	X	
Augmented Reality			X
Computer Vision			X
Requirements Engineering und Management		X	
Big Data	X	X	
E-Business-Management	X	X	
Identity und IT-Architekturmanagement		X	
eHealth		X	
Datenschutz		X	
Kryptographie			X
Integration und Management von Geschäftsprozessen mit SAP		X	
Data Science	X		

- 1) Data Science
- 2) Wirtschaftsinformatik und E-Business
- 3) Autonome Systeme und Technische Informatik

Studiengang in Teilzeit

Informatik, Master (M.Sc.) - Teilzeit Hochschule für angewandte Wissenschaften Kempten gültig mit Studienbeginn WS 2017/18

Semester	gültig mit Studienbeginn WS 2017/18														
6	Masterarbeit ⁽⁶⁾														
5	Masterarbeit ⁽⁶⁾														
4	Projekt ⁽⁴⁾	WP-Fach ⁽⁵⁾	WP-Fach ⁽⁵⁾												
3	Hauptseminar ⁽⁴⁾	WP-Fach ⁽⁵⁾	WP-Fach ⁽⁵⁾												
2	WP-Fach ⁽⁵⁾	WP-Fach ⁽⁵⁾	WP-Fach ⁽⁵⁾												
1	Effiziente Algorithmen ⁽³⁾	Mathematik ⁽¹⁾	Softwarearchitektur ⁽²⁾												
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

ECTS-Punkte

Legende:

1 Mathematik	4 Angewandte Informatik und interdisziplinäre Fächer
2 Softwaretechnik	5 Wahlfach
3 Theorie der Informatik	6 Seminar- und Abschlussarbeit



Für den Studiengang Informatik (Master) stehen folgende Ansprechpartner zur Verfügung:

Studiengangkoordinator: Prof. Dr. Elmar Böhler
Fachstudienberater: Prof. Dr. Georg Hagel
Vorsitzender der Prüfungskommission: Prof. Dr. Ulrich Göhner



2 Begriffserläuterungen

ECTS - European Credit Transfer System

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

Arbeitsaufwand (Workload) und Leistungspunkte (ECTS-LP)

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

Der Arbeitsaufwand setzt sich zusammen aus:

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

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

Semesterwochenstunden und Präsenzzeit

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

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



Module und Wahlpflichtmodule

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

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

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

Die Lehrveranstaltungen werden derzeit in deutscher Sprache gehalten.

Studienbegleitende Prüfungen und Studienfortschritt

Sämtliche Prüfungen erfolgen über das gesamte Studium verteilt studienbegleitend und stehen in direktem Bezug zur Lehrveranstaltung. Prüfungsbestandteile können je nach Lehrveranstaltung begleitend oder nach Abschluss des Moduls stattfinden, beispielsweise als Referat, Klausurarbeit, mündliche Prüfung, Hausarbeit mit Kolloquium, Entwurf mit Kolloquium, Laborbericht, Exkursionsbericht oder einer Kombination. In den Beschreibungen der einzelnen Module wird im Modulhandbuch die jeweilige Prüfungsform festgelegt. Eine Wiederholung der Prüfung eines Moduls erfolgt bei Nichtbestehen im folgenden Semester. Die Prüfung für ein Modul darf in der Regel nur einmal wiederholt werden, genau regelt dies die Rahmenprüfungsordnung.

Zur Masterarbeit kann sich anmelden, wer mindestens 35 ECTS-Punkte erreicht und in mindestens 4 Pflichtfächern die Note "ausreichend" oder besser erzielt hat.

3 Modulbeschreibungen

IFM1101 Effiziente Algorithmen

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Ulrich Göhner
Dozent(en):	Prof. Dr. Ulrich Göhner / Prof. Dr. Jochen Staudacher
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Mündliche Prüfung 30 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden kennen die Begriffe der Graphentheorie
- Sie können wichtige Algorithmen zur Lösung graphentheoretischer Probleme anwenden und die Komplexität der Algorithmen und Methoden bewerten
- Sie können die gelernten Methoden auf praktische Beispiele anwenden
- Sie kennen die zentralen Methoden der Komplexitätstheorie und haben Verständnis für die Relevanz komplexitätstheoretischer Betrachtungen in der Praxis
- Sie können einfache Spiele klassifizieren und kennen den Nutzen spieltheoretischer Ansätze und Methoden in der Informatik
- Sie kennen Methoden und spieltheoretische Modelle zur Beschreibung von Auktionsverfahren und können diese auf einfache Fragestellungen aus der Internet-Ökonomie anwenden
- Sie haben Einblick in Funktionsweise, Grenzen und Möglichkeiten randomisierter Algorithmen
- Sie haben Einblick in Funktionsweise, Grenzen und Möglichkeiten des Prinzips der dynamischen Programmierung

Lehrinhalte:

- Begriffe und Definitionen der Graphen-Theorie
- Zusammenhang

- Topologisches Sortieren
- Algorithmen zur Bestimmung des kürzesten Wegs (A*-, Dijkstra-, Floyd- und Warshall-Algorithmus)
- Lokale Suchalgorithmen (z.B. Bergsteiger- und Ameisen-Algorithmus)
- Bestimmung des minimalen Spannbaums mit Algorithmen von Prim und Kruskal
- Bestimmung des maximalen Flusses in Netzwerken mit den Algorithmen von Ford/Fulkerson, Edmonds/Karp und Dinic
- Lösung des Maximum-Matching Problems mit Hilfe von Netzwerken
- Praktischer Einsatz von Algorithmen auf Graphen mit BGL Boost
- Die Komplexitätsklassen P und NP
- NP-vollständige Probleme, insbesondere auch aus der Graphentheorie
- Grundelemente der Algorithmischen Spieltheorie
- Stabile Matchings, Mechanismen, Auktionen und deren praktische Anwendung in der Internet-Ökonomie
- Komplexität von Spielen
- Randomisierte Algorithmen
- Prinzipien der dynamischen Programmierung

Literatur:

- Logofatu, D.: "Algorithmen und Problemlösungen mit C++", Vieweg
- Heun, V.: "Grundlegende Algorithmen", Vieweg.
- Saake, G; Sattler, U.: "Algorithmen und Datenstrukturen", Dpunkt-Verlag
- Krumke, S, Noltemeier, H.: "Graphentheoretische Konzepte und Algorithmen", Teubner
- Gelperin, D.: "On the optimality of A* Artificial Intelligence, 8:69-76, 1977.
- Sedgewick, R. und J. S. Vitter: "Shortest paths in Euclidean graphs." *Algorithmica*, 1(1):31-48, 1986.
- S. Arora, B. Barak: *Computational Complexity*, Cambridge University Press, 1. Auflage, 2009.
- J.E. Hopcroft, R. Motwani, J.D. Ullman: *Einführung in Automatentheorie, Formale Sprachen und Berechenbarkeit*, Pearson Studium, 3. Auflage, 2011.
- M. Mitzenmacher, E. Upfal: *Probability and Computing*, Cambridge University Press, 2. Auflage, 2017.
- M.J. Osborne, A. Rubinstein: *A Course in Game Theory*, MIT Press, 1. Auflage, 1994.
- H. Peters: *Game Theory*, Springer, 2. Auflage, 2015.
- M. Maschler, E. Solan, S. Zamir: *Game Theory*, Cambridge University Press, 1. Auflage, 2013.
- Y. Saez, A. Mochon: *Understanding Auctions*, Springer, 1. Auflage, 2015.
- S.R. Chakravarty, M. Mitra, P. Sarkar: *A Course on Cooperative Game Theory*, Cambridge University Press, 2015

IFM1102 Mathematik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Markus Preisenberger
Dozent(en):	Prof. Dr. Markus Preisenberger / Epple
Modultyp:	Pflichtmodul
Voraussetzungen:	Analysis einer Veränderlichen, Lineare Algebra
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht/Übung 30 Stunden Selbststudium - Betreute Studierzeit 60 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters Mathematik-Onlinetest als Zulassungsvoraussetzung
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme am Modul Mathematik können die Studierenden

- zentrale Konzepte der Theorie linearer und nichtlinearer gewöhnlicher Differentialgleichungen wiedergeben
- komplexe Aufgabenstellungen hinsichtlich der Lösbarkeit mit Methoden der Theorie dynamischer Systeme analysieren
- geeignete Lösungsmethoden aus verschiedenen Teilgebieten der Mathematik auswählen und zu einem Lösungsansatz für Aufgaben aus dem Bereich dynamischer Systeme kombinieren

Lehrinhalte:

- Komplexe Zahlen, algebraische Operationen und deren geometrische Interpretation
- Komplexwertige Funktionen, Kurven in der komplexen Ebene, Grenzwerte, Differentiation und Integration komplexwertiger Funktionen
- Gewöhnliche lineare Differentialgleichungen; Homogene und inhomogene Gleichungen; Lösungsbegriff, reelle und komplexe Lösungen, Existenz und Eindeutigkeit; Lineare Differentialoperatoren, Lösungsraum, charakteristisches Polynom, Basisfunktionen; Partikuläre Lösung und Lösungen inhomogener Gleichungen; Differentialgleichungssysteme; Phasendiagramm und qualitative Eigenschaften der Lösungen
- Gewöhnliche nichtlineare Differentialgleichungen Beispiele; Phasendiagramm; Linearisierung; Stationäre, stabile und instabile Lösungen
- Fourier-Reihen, Fourier-Transformation; Definition, Konvergenz, Eigenschaften, DFT, FFT, Bandbegrenzte Funktionen, Anwendungsbeispiele

- Weiterer Anwendungsbereich

Literatur:

- Butz, T.: Fourier-Transformation für Fußgänger, B.G. Teubner, 2. Auflage
- Hachenberger, D.: Mathematik für Informatiker, Pearson Studium, 2. Auflage, 2008
- Hirsch, M.W., Smale, S.: Differential Equations, Dynamical Systems and Linear Algebra Academic Press, 1974
- Lang, H. W.: Algorithmen in Java, Oldenbourg Verlag, 2. Auflage, 2006
- Stingl, P.: Mathematik für Fachhochschulen, Hanser Verlag, 8. Auflage, 2009



IFM1103 Softwarearchitektur

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Georg Hagel
Dozent(en):	Prof. Dr. Georg Hagel
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweis in der Übung, Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

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

- Zu erklären, welche Faktoren Einfluss auf die Architektur einer Software haben.
- Zu beschreiben, welche Architekturprinzipien beim Entwurf einer Softwarearchitektur beachtet werden müssen.
- Bekannte Architekturmuster bei der Erstellung der Softwarearchitektur anzuwenden.
- Aspekte von Softwaresystemen zu beschreiben.
- Lösungsansätze für Aspekte von Softwaresystemen anzugeben.
- Softwarearchitekturen mit aktuellen Beschreibungsmethoden zu dokumentieren.
- Continuous Integration, Continuous Deployment und DevOps zu beschreiben und anzuwenden.

Lehrinhalte:

- Aufgaben von Softwarearchitekten
- Entwurf von Softwarearchitekturen
- Architekturmuster
- Architektur Aspekte
- Dokumentation von Softwarearchitekturen
- Continuous Integration, Continuous Deployment und DevOps

Literatur:

-
- Posch, Thorsten et al.: Basiswissen Softwarearchitektur, dpunkt.verlag, 3. Auflage (2012)
 - Starke, Gernot: Effektive Software-Architekturen, Carl Hanser Verlag, 8. Auflage (2017)
 - Vogel, Oliver et al.: Software-Architektur, Elsevier (2005)
 - Buschmann, Frank et al.: POSA Band 1 - Pattern-orientierte Softwarearchitektur - Ein Pattern-System
 - Bass, Len et al.: Software Architecture in Practice, Third Edition, SEI Series in Software Engineering, Addison-Wesley (2012)
 - Oestereich, Bernd: Die UML 2.5 Kurzreferenz für die Praxis, Oldenbourg Verlag, 6. Auflage (2014)
 - Bass, Len et al.: DevOps: A Software Architect's Perspective, Addison Wesley (2015)



IFM5201 Data Science

Allgemeines

Data Science ist ein interdisziplinäres Wissenschaftsfeld mit Schnittmengen zur Informatik und zur Statistik und beschäftigt sich mit der Extraktion von Wissen aus Daten. Das Ziel ist es, Erkenntnisse und Handlungsempfehlungen in konkreten Problemstellungen zu generieren und zugleich, wo möglich, das Maß der Unsicherheit in den Erkenntnissen zu quantifizieren. Angesichts der sprunghaft wachsenden Datengenerierung in allen Bereichen der modernen Gesellschaft findet Data Science vielzählige kommerzielle Einsatzmöglichkeiten, beispielsweise im Kontext Industrie 4.0, insbesondere, aber nicht ausschließlich, in Big Data Szenarien. Die Veranstaltung geht auf Grundlagen der Statistik und des Datenmanagements ein, erklärt Methoden des Maschinellen Lernens, betont die Rolle der Datenvisualisierung und des Storytellings und versucht darüberhinaus das Verständnis für typische Fallstricke und allgemeiner für die Grenzen der Dateninterpretation zu schärfen.

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

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage, in einem Unternehmen im Gespräch mit einem Mitarbeiter der Führungs- oder Fachebene eine datenbezogene Problemstellung zu analysieren, die relevanten Daten zu identifizieren, diese in einer geeigneten Infrastruktur zu erfassen, weiterzuverarbeiten und in nachfolgenden Analyseschritten Handlungsempfehlungen zu erarbeiten. Hierzu wird Python in Verbindung mit Jupyter Notebook eingesetzt. Es werden grundlegende Verfahren der Statistik und des Maschinellen Lernens beherrscht. Bei ungenügender Datenqualität oder -quantität sind die Studierenden befähigt, zu argumentieren, dass keine valide Aussage getroffen werden kann.

Lehrinhalte:

- Projektmanagement im Data Science-Umfeld (insb. CRISP-DM)

- Deskriptive Statistik (Mittelwert, Median, Pearson- und Spearman-Korrelation, Quantile, ...)
- Statistische Tests (u-Test, t-Test)
- Versuchsplanung (Design of Experiment)
- Datenvisualisierung und Storytelling
- Grundlagen des Datenmanagements
- Datenreinigung und Feature Engineering
- Methoden des Maschinellen Lernens (supervised und unsupervised)
- Signalanalyse und Anomaliedetektion
- Python und Jupyter Notebook mit gängigen Paketen wie numpy, pandas, scikit-learn oder matplotlib

Literatur:

- Jake VanderPlas: Data Science Handbook (online verfügbar)
- Wes McKinney: Python for Data Analysis
- Christopher M. Bishop: Pattern Recognition and Machine Learning
- J. Grus: Einführung in Data Science
- S. Russell, P. Norvig: Artificial Intelligence

IFM5301 Betriebssystem im Eigenbau

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Frenz
Dozent(en):	Prof. Dr. Stefan Frenz
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweise zu den Übungen, mündliche Prüfung 30 Minuten am Ende des Semesters, Leistungsnachweise sind Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden verstehen die grundlegenden Herausforderungen an der Implementierung von Betriebssystemen.
- Sie kennen die im Umfeld von aktuellen 32 und auch 64 Bit Rechnern relevanten praktischen sowie die erforderlichen theoretischen Hintergründe.
- Sie verstehen das Zusammenspiel zwischen Compiler, Laufzeitumgebung und Betriebssystem.

Lehrinhalte:

- x86-Architektur in Bezug zur Betriebssystementwicklung
- Design-Kriterien für maßgeschneiderte Betriebssysteme
- Gerätetreiber im allgemeinen und speziellen
- Anforderungen an Laufzeitsysteme
- Compiler-Grundlagen

Literatur:

- Tanenbaum, Andrew S.: Moderne Betriebssysteme (3. Auflage, 2009)
- Wirth, Niklaus; Gutknecht, Jürg: Project Oberon (2005)

IFM5302 Echtzeitsysteme

Allgemeines

Die Lehrveranstaltung führt in das Thema Echtzeitsysteme ein und gibt einen Überblick über spezifische Problemstellungen und Lösungen im Echtzeitbetrieb.

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Programmierkenntnisse (bevorzugt in C oder C++)
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden kennen Beispiele von Echtzeitsystemen
- Sie verstehen den Begriff "Echtzeit" richtig zu deuten und können verschiedene Klassen von Echtzeitsystemen unterscheiden
- Sie kennen die Grundlagen des Prozessscheduling und wissen um die Besonderheiten der Anforderungen des Scheduling bei Echtzeitbetriebssystemen
- Sie kennen verschiedene echtzeitfähige Schedulingverfahren und ihre jeweiligen Vor- und Nachteile
- Sie wissen um die Besonderheiten der Speicherverwaltung bei Echtzeitsystemen
- Die Studierenden verstehen das Problem der Prioritätsinversion, das bei Echtzeitsystemen mit mehreren rechnenden voneinander abhängigen Tasks auftreten kann
- <p>Sie kennen Beschleunigungsverfahren, um Algorithmen "echtzeitfähig" zu machen</p>

Lehrinhalte:

- <p>Beispiele für Echtzeitsysteme</p>
- <p>Genauere Definition des Begriffs "Echtzeit"</p>
- <p>Klassifikation von Echtzeitsystemen</p>
- <p>Grundlagen der (Echtzeit-)betriebssysteme</p>
- <p>Besonderheiten von Echtzeitbetriebssystemen hinsichtlich Prozessscheduling und Speicherverwaltung</p>

- Das Prioritätsinversionsproblem
- Verfahren zur Beschleunigung von Algorithmen

Literatur:

- Jane W.S. Liu: Real-Time Systems. Pearson-Verlag. 2000.
- Dieter Zöbel: Echtzeitsysteme. Grundlagen der Planung. Springer-Verlag. 2008.
- Jürgen Quadde, Michael Mächtel: Moderne Realzeitsysteme kompakt: Eine Einführung mit Embedded Linux: Praxis und Theorie mit Embedded Linux. dpunkt-Verlag. 2012.
- Thilo Streichert, Matthias Traub: Elektrik/Elektronik-Architekturen im Kraftfahrzeug: Modellierung und Bewertung von Echtzeitsystemen (VDI-Buch). Springer-Vieweg-Verlag. 2012



IFM5303 Multimodale Sensorsysteme

Allgemeines

Die Lehrveranstaltung befasst sich mit der Verarbeitung von Sensordaten - insbesondere der Fusion von Sensordaten, die aus mehreren (auch unterschiedlichen) Sensordatenquellen stammen. Dies können zum Beispiel mehrere Kamerabilder sein, die zu einer Objektdetektion fusioniert werden. Als grundlegende Techniken werden vor allem der rekursive Bayessche Schätzer als etablierter Zustandsschätzer sowie lernende Fusionsarchitekturen, wie das Convolutional Neural Network, behandelt.

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Lineare Algebra, Programmierkenntnisse
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden wissen wieso die Sensordatenfusion in realen multi-modalen Sensorsystemen notwendig ist und können Fusionsansätze klassifizieren
- Sie kennen und verstehen den probabilistischen Hintergrund des Zustandsschätzproblems
- Sie kennen eine konkrete Realisierung des rekursiven Bayesschen Schätzer durch den Partikel-Filter
- Sie kennen und verstehen neuere Architekturen zur Sensordatenfusion wie Convolutional Neural Networks (CNNs) und wie diese für konkrete Aufgaben wie die Objektdetektion eingesetzt werden können

Lehrinhalte:

- Warum ist Sensordatenfusion wichtig?
- Klassifikation von Fusionsansätzen
- Probabilistischer Rahmen des Zustandsschätzproblems: der rekursive Bayessche Filter
- Realisierung des rekursiven Bayesschen Schätzers über Partikel-Filter
- Repräsentation von Sensordaten in hierarchischen Fusionsarchitekturen wie CNNs

zur Objektdetektion

- Representation Learning

Literatur:

- Francois Chollet. Deep Learning with Python. Manning, 2018.



IFM5304 Mobile Roboter

Allgemeines

Mobile Roboter sind immer noch nicht in unserem Alltag allgegenwärtig. Zwar gibt es einige wenige Beispiele wie Staubsaugroboter, Drohnen, Marserkundungsroboter, autonome Fahrzeuge oder Fußballspielende Roboter beim RoboCup, doch meist handelt es sich bisher um Nischenlösungen. Wieso ist es so schwer mobile Roboter zu bauen, die Arbeiten übernehmen können, die heute nur von Menschen ausgeführt werden können? In der Vorlesung soll ein breiter Überblick über die verschiedenen Aspekte des Themas mobile Roboter erarbeitet werden. Nachdem ein Verständnis für die vielseitigen Herausforderungen der mobilen Robotik geschaffen wird, werden wir auf geeignete Sensorik und Aktorik für Roboter eingehen. Die Organisation der Informationsverarbeitung ist in einem mobilen Roboter ein wichtiges zentrales Thema. Entsprechend werden hier verschiedene Ansätze für sog. Roboterkontrollarchitekturen vorgestellt. Ein ebenso zentrales Thema ist das Thema der Navigation: wie kann ein mobiler Roboter robust durch eine bekannte oder sogar unbekannte Umgebung navigieren? Schließlich behandeln wir das Thema des Lernens von geeigneten Aktionen für Roboter. Hier haben sich verschiedene Ansätze wie z.B. das Reinforcement-Learning und das Imitationslernen etabliert. Am Ende der Vorlesung werden wir auf Besonderheiten der kooperativen Robotik eingehen und es werden Ansätze vorgestellt, die eine Zusammenarbeit von mehreren Robotern ermöglichen sollen. In den Übungen sollen zentrale Algorithmen anhand von Robotersimulatoren vertieft werden.

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Analysis, Lineare Algebra, Programmierkenntnisse
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweise in den Übungen Leistungsnachweise sind Zulassungsvoraussetzungen zur schriftlichen Prüfung Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden können einschätzen, welche Fähigkeiten heute von mobilen Robotern erwartet werden können und welche nicht
- Sie wissen welche Sensorik für eine robuste Umfelderkennung notwendig ist und



kennen Ansätze zur Umfelderkennung auf Basis geeigneter Sensordaten

- Sie haben fundiertes Wissen über verschiedene Ansätze von Roboterkontrollarchitekturen und kennen deren Vor- und Nachteile
- Die Studierenden kennen unterschiedliche Navigationsverfahren und können diese selber implementieren und anwenden
- Sie haben einen breiten Überblick über Verfahren die das Erlernen von Aktionen erlauben, kennen den theoretischen Hintergrund der Verfahren und können die Verfahren selber umsetzen
- Den Studierenden sind mögliche Auswirkungen der Verfügbarkeit intelligenter mobiler Roboter auf die Gesellschaft bewusst und können kritisch beurteilt werden

Lehrinhalte:

- Einführung: Geschichte der mobilen Roboter
- Herausforderungen für mobile Roboter
- Roboterkontrollarchitekturen
- Navigation, Lokalisation und Erstellung von Karten (SLAM), Pfadplanung
- Lernen von Aktionen, Reinforcement-Learning, Imitationslernen
- Kognitive Architekturen für Roboter
- Ausblick: wohin entwickelt sich der Bereich mobile Roboter?
- Mögliche Auswirkungen der Verfügbarkeit intelligenter mobiler Roboter auf die Gesellschaft

Literatur:

- Joachim Hertzberg et al. Mobile Roboter: Eine Einführung aus Sicht der Informatik Springer Vieweg Verlag. 2012.
- Roland Siegwart et al. Introduction to Autonomous Mobile Robots MIT. 2011



IFM5309 Kryptographie

Allgemeines

Wir beschäftigen uns mit verschiedenen gängigen Kryptographieverfahren (siehe Themenliste) sowie grundsätzlichen Fragestellungen rund um die Kryptographie: Woher kommt die Kryptographie? Was ist nötig um Kryptographie zu betreiben? Wann ist sie sicher? Wie wirkt sich Kryptographie (bzw. ihr Abwesenheit) auf die moderne Gesellschaft aus? Wie können Wahlen sicher durchgeführt werden? Wie können Kryptosystem analysiert/gebrochen werden? Welche Entwicklungen, die die Kryptographie wesentlich verändern werden, sind mittelfristig zu erwarten (Stichwort Quantencomputer)? Was gibt es für Ansätze um dem entgegenzuwirken?

Modulverantwortliche(r):	Prof. Dr. Elmar Böhler
Dozent(en):	Prof. Dr. Elmar Böhler
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Programmieren I+II, Algorithmen und Datenstrukturen, Theoretische Informatik, Diskrete Mathematik, Wahrscheinlichkeitsrechnung
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, einseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Lehrinhalte:

Es werden die Grundlagen der Kryptographie behandelt, sowie einige der wesentlichsten verwendeten Verfahren. Mögliche Themen sind die folgenden:

- Historische Kryptosysteme und ihre Analyse
- Blockchiffren
- Betriebsarten
- DES
- AES

- Schlüsselaustausch
- Public-Key Kryptographie
- RSA
- Digitale Signaturen
- Primzahltests
- ElGamal
- Einführung in ECC
- Teilen von Geheimnissen
- Einblicke in die Quantenkryptographie
- Postquantenkryptographie
- Hashfunktionen
- Wahlverfahren
- Blockchains
- Pseudozufallszahlen
- Kryptographie in der Gesellschaft
- Zero-Knowledge Systeme
- Homomorphe Verschlüsselung
- Komplexitätstheoretische Grundlagen
- Berechnungsverifikation

Literatur:

- J. Buchmann: Einführung in die Kryptographie
- J. Rothe: Complexity Theory and Cryptology
- K. Wagner: Theoretische Informatik
- A. Menezes, P. van Oorschot, S. Vanstone: Handbook of Applied Cryptography

IFM5311 Computer Vision

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Klaus Ulhaas
Dozent(en):	Prof. Dr. Klaus Ulhaas
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Englischkenntnisse
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage

- den theoretischen Hintergrund ausgewählter Verfahren der Computer Vision und der OpenCV zu erklären.
- die gelernten Verfahren zu kombinieren und mit der OpenCV in C++ auf Beispiele aus der Praxis zu übertragen.
- die projektive Geometrie für Ein- und Zweikerasysteme zu verstehen, sowie die grundlegenden Schritte zur Triangulation wiederzugeben und zu berechnen

Lehrinhalte:

Folgende Inhalte werden in der Veranstaltung bearbeitet und an praktischen Beispielen implementiert:

- Einführung in OpenCV
- 2D Computer Vision:
 - Segmentierung,
 - Filteroperationen,
 - Objekterkennungsmethoden
 - Merkmalerkennung
 - Homographie-Abbildungen
- 3D Computer Vision:
 - Kamerakalibration,



- 3D Szenengeometrie

Literatur:

Literatur und Empfehlungen zu Einzelthemen werden fortlaufend in der Vorlesung bekanntgegeben

- age, 2014.



IFM5401 Requirements Engineering und Management

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Georg Hagel
Dozent(en):	Prof. Dr. Georg Hagel
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht - Just-in-Time-Teaching - Feedback durch den Einsatz automatischer Antwortsysteme 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters, Leistungsnachweise in der Veranstaltung, Leistungsnachweise sind Zulassungsvoraussetzungen.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

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

- die Aufgaben von Requirements-Engineers zu beschreiben.
- Techniken, mit denen funktionale und nicht-funktionale Anforderungen erhoben und dokumentiert werden zu beschreiben.
- Techniken, mit denen funktionale und nicht-funktionale Anforderungen erhoben und dokumentiert werden, anzuwenden.
- qualitätssichernde Regeln des SOPHIST-Regelwerks auf bestehende Anforderungen anzuwenden.
- funktionale und nicht-funktionale Anforderungen für Softwaresysteme zu unterscheiden, und qualitätszusichern.
- durch die Lehr- und Lernform "Just-in-Time-Teaching" selbstständig Lerninhalte zu erarbeiten - eine für das lebenslange Lernen notwendige Technik.
- wissenschaftliche Kurzvorträge zu erstellen und zu halten.

Lehrinhalte:

- Der Requirements-Engineer
- Anforderungsermittlung
- Das SOPHIST-Regelwerk
- Schablonen für Anforderungen



-
- Dokumentation von Anforderungen
 - Nicht-funktionale Anforderungen

Literatur:

Chris Rupp et al.: Requirements-Engineering und -Management, Hanser, 6. Auflage (2014)



IFM5402 Identity- und IT-Architekturmanagement

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Peter Klutke
Dozent(en):	Prof. Dr. Peter Klutke
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Praktikum in kleinen Projektteams
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Mündliche Prüfung 20 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

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

- die Grundlagen des Identitymanagements und des IT-Architekturmanagements strukturiert darzustellen und zu erklären,
- dazugehörige Konzepte und Strategien zu vergleichen, zu beurteilen und auf neue Situationen anzuwenden und
- gesellschaftliche Auswirkungen und Sozialverträglichkeit von Lösungen und Innovationen besser zu beurteilen, etwa im Identity Management bei der Authentifizierung und Autorisierung in Notsituationen oder im Enterprise Architecture Management beim mehrdimensionalen Denken betreffend das "Enterprise Intergrating" und die "Enterprise Ecological Adaptation".

Lehrinhalte:

- Grundlagen des Identitymanagements mit Einführung, Begriffen, einem Ebenenmodell, zugehörigen Prozessen und Technologien, Rollenmodellen sowie organisationsübergreifendem Identitymanagement (Federation)
- Grundlagen des IT-Architekturmanagements mit Einführung, Begriffen, Bedeutung, Bestandteilen, der Architekturpyramide und deren Schichten, Modellen zur Architekturplanung, Referenzarchitekturen sowie dazugehörigen Prozessen
- Besprechung von Fallbeispielen zur Veranschaulichung im Praktikum unter Einsatz aktivierender Lernmethoden.

Literatur:

- Alexander Tsolkas, Klaus Schmidt: "Rollen und Berechtigungskonzepte: Identity- und Access-Management im Unternehmen" , Verlag: Springer Vieweg; Auflage: 2. Aufl. 2017 (14. Juli 2017), ISBN-13: 978-3658179861

-
- Christian Mezler-Andelberg: „Identity Management - eine Einführung“; dpunkt.verlag; 1. Aufl. (Okt. 2007)
 - Inge Hanschke: "Enterprise Architecture Management - einfach und effektiv: Ein praktischer Leitfaden für die Einführung von EAM", Verlag: Carl Hanser Verlag GmbH & Co. KG; Auflage: 2., überarbeitete (8. August 2016), ISBN-13: 978-3446447240
 - Gernot Dern: "Management von IT-Architekturen: Leitlinien für die Ausrichtung, Planung und Gestaltung von Informationssystemen"; Vieweg+Teubner Verlag; 3., durchges. Aufl. (März 2009)

IFM5403 eHealth

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Rafael Mayoral Malmström
Dozent(en):	Prof. Dr. Rafael Mayoral Malmström
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Die erfolgreiche Teilnahme an der Übung gilt als Zulassungsvoraussetzung für die 90 minütige schriftliche Prüfung am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Nach erfolgreicher Beendigung der Veranstaltung haben die Studierenden einen Überblick über die aktuelle Situation der Gesundheitsversorgung und können die Notwendigkeit einer IT-Unterstützung der Prozesse im Gesundheitswesen erklären.
- Die Studierenden kennen verfügbare und zukünftige Anwendungen von eHealth, um optimal Geschäfts- und klinische Prozesse im Gesundheitswesen zu unterstützen.
- Die Studierenden sind in der Lage die Anforderungen, inklusive regulatorisch, an die IT-Systeme im Rahmen des Gesundheitswesens zu verstehen.
- Die Studierenden haben einen Überblick über aktuelle und zukünftige Entwicklungen der IT im Gesundheitswesen.

Lehrinhalte:

- Grundlagen des Gesundheitswesens und Motivation für den Einsatz von IT-Systemen
- Ziele und Einsatzbereiche von eHealth
- Beispielanwendungen von eHealth: eGK, eRezept, eMedikation, Gesundheitsportal, Telemedizin, eDokumentation
- Gesundheitstelematik: Akteure, Komponenten, Protokolle
- Entwicklung von eHealth in Deutschland: eHealth-Gesetz, regulatorischer Rahmen
- Aktuelle Entwicklungen: data-driven health, precision medicine, mHealth

- Blick ins Ausland: ehealthsuisse, Healthcare Denmark, ELGA (Österreich)

Literatur:

- Jähn, K. Und Nagel, E.: "eHealth", Springer (2014), 978-3642639319
- Bauer, Eickmeier, Eckard: "E-Health: Datenschutz und Datensicherheit: Herausforderungen und Lösungen im IoT-Zeitalter", Springer (2017), 978-3658150907
- Fischer, Krämer: "eHealth in Deutschland: Anforderungen und Potenziale innovativer Versorgungsstrukturen", Springer (2016), 978-3662495032
- Trill, R. (Hrsg.): "Praxisbuch eHealth", Kohlhammer (2009), 978-3-17-019988-0
- Haas, P.: "Gesundheitstelematik", Springer (2006), 978-3540207405
- Aktuelle Publikationen zu den oben genannten Themen



IFM5404 Datenschutz

Allgemeines

Datenschutz - von hoher gesellschaftlicher Relevanz, nicht nur für Informatiker. Studenten werden an Hand konkreter Beispiele für die Notwendigkeit des verantwortungsvollen Umgangs mit personenbezogenen Daten sensibilisiert. Hier wird der Bogen vom "right to be left alone" aus der amerikanischen Unabhängigkeitserklärung über Art. 2, Abs. 1 in Verbindung mit dem zentralen Art. 1, Abs. 1 GG zur EU-DSGVO geschlagen. Dann wird die praktische Umsetzung im Rahmen eines KUM besprochen und im Rahmen unserer Hochschule angewandt.

Modulverantwortliche(r):	Prof. Dr. Arnulf Deinzer
Dozent(en):	Prof. Dr. Arnulf Deinzer
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen (wöchentlich 90 Minuten)
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Anwesenheitspflicht in den Übungen, schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweis ist Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfsmittel:	Gesetzestexte mit handschriftlichen Anmerkungen

Lernergebnisse und Inhalte

Lernergebnisse:

- Kenntnisse, die zur Einarbeitung in die Aufgaben eines betrieblichen oder behördlichen Datenschutzbeauftragten (DSB) nötig sind

Lehrinhalte:

- Informationstechnik und Persönlichkeitsrecht
- Datenschutz-, Computer- und Arbeitsrecht
- IT-Sicherheit
- Praxis des Datenschutzes

Literatur:

- Bergmann, Möhrle, Herb: Datenschutzrecht, Boorberg 56. AL 11/2018
- Wilde, Ehmman, Niese, Knoblauch: Bayerisches Datenschutzgesetz, 30. AL 11/2018



IFM5405 Integration und Management von Geschäftsprozessen mit SAP

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Wind
Dozent(en):	Prof. Dr. Stefan Wind
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Die erfolgreiche Teilnahme an der Übung gilt als Zulassungsvoraussetzung für die 90 minutige schriftliche Prüfung am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden

- kennen moderne Integrationsplattformen und Integrationsszenarien
- sind in der Lage, komplexe, integrierte Geschäftsprozesse zu analysieren, zu beschreiben und zu modellieren
- sind durch ein grundsätzliches Verständnis befähigt, integrierte Informationssysteme am Beispiel von SAP ERP Systemen zu verstehen und zu beschreiben
- können diese Kenntnisse auf die Integration und das Management der grundlegenden Geschäftsprozesse in den Bereichen Kundenauftragsmanagement, Beschaffung, Finanzwesen usw. entlang der horizontalen und vertikalen Integration in SAP ERP-Systemen transferieren

Lehrinhalte:

- Grundlagen der Integration und Management von Geschäftsprozessen mit SAP
- Moderne Integrationsplattformen und Szenarien (z. B. SAP NetWeaver)
- Integrierte Informationssysteme am Beispiel von ERP und In-Memory-basierte ERP-Systemen von SAP
- Modellierung von integrierten Geschäftsprozessen und Geschäftsprozessmanagement mit ARIS
- Integrierte Geschäftsprozesse und prozessorientierte Integration der betrieblichen Informationsverarbeitung in SAP

- Fallstudien zu komplexen integrierten Geschäftsprozessen und Informationssystemen mit SAP ERP GBI

Literatur:

- Mertens, P.: Integrierte Informationsverarbeitung 1 - Administrations- und Dispositionssysteme in der Industrie, 18. Aufl., Wiesbaden 2013
- Mertens, P., Meier M.: Integrierte Informationsverarbeitung 2 - Planungs- und Kontrollsysteme in der Industrie, 10. Aufl., Wiesbaden 2009
- Scheer, A.-W.: Wirtschaftsinformatik - Referenzmodelle für industrielle Geschäftsprozesse, Studienausgabe, 2. Aufl., Berlin, Heidelberg, New York 1998
- Scheer, August-Wilhelm. ARIS-vom Geschäftsprozess zum Anwendungssystem. Springer-Verlag, 2013



IFM5406 Funktionale Sicherheit

Allgemeines

Einführung in Funktionale Sicherheit mit besonderer Beachtung der Anforderungen von Informatik, Fahrerassistenzsystemen und Automatisierung & Robotik

Modulverantwortliche(r):	Prof. Dr. Rolf Jung
Dozent(en):	Prof. Dr. Rolf Jung / Prof. Dr. Stefan-Alexander Schneider
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	Übung/Praktikum findet während des Semesters in Blöcken statt.
Leistungspunkte:	5
Arbeitsaufwand:	52,5 Stunden Präsenzzeit Unterricht 7,5 Stunden Präsenzzeit Praktikum/Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	Benutzung von Teilen der Normen ISO 26262 und ISO 13849 nach Vorgabe, nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

- Studenten kennen den Begriff 'Funktionale Sicherheit' und damit verbundene Definitionen aus den Normen
- Teilnehmer kennen die Elemente des Sicherheitsprozesses und können sie dem Entwicklungsprozess zuordnen
- Sie kennen Managementmethoden und können diese dem Sicherheitslebenszyklus zuordnen
- Studenten können die Methode der HARA anwenden
- Teilnehmer können zwischen reparierbaren und nicht reparierbaren Elementen unterscheiden und die Funktionen Unzuverlässigkeit und Nichtverfügbarkeit technischer Elemente erklären
- Studenten kennen die unterschiedlichen logischen Operatoren der Wahrscheinlichkeitskombinatorik und kennen die Modellbildung in der FTA bzw. anderen Sicherheitsanalysen
- Studenten können die Methode FMEDA im Sicherheitsprozess einordnen sowie Sicherheitskennzahlen bestimmen und anwenden
- Teilnehmer kennen Testmethoden bezüglich System, Hardware, Software und können aus System- und Produkthanforderungen Testanforderungen erstellen



- Studenten können die Anforderungen der Norm ISO26262 zum Produktionsprozess im Sicherheitslebenszyklus anwenden
- Sie kennen den Begriff der Zugangssicherheit im Kontext von Funktionaler Sicherheit
- Studenten können Softwarewerkzeuge klassifizieren
- Die Studenten kennen einen Zulassungsprozess und können Dokumente für eine Sicherheitszulassung zusammenstellen

Lehrinhalte:

- Einführung in die Gesetzeslage und Normenüberblick mit Begriffserklärungen
- Erläuterung der Elemente eines Sicherheitsprozesses und Erstellung eines Sicherheitsplans und Sicherheitsnachweises
- Analyse der Elemente und Schritte in einem Sicherheitslebenszyklus
- Durchführung einer Risiko- und Gefährdungsanalyse
- Ermittlung von Sicherheitskennzahlen an einem Fallbeispiel
- Einführung in mathematische Methoden der Zuverlässigkeitstheorie und Durchführung einer Sicherheitsanalyse für Fehlerkombinationen
- Erarbeitung einer FMEDA mit Fallbeispiel
- Kennenlernen von Testmethoden und Erstellen eines V&V-Plans
- Einführung in Zugangssicherheit
- Klassifizierung von Softwarewerkzeugen entsprechend Sicherheitsnormen

Literatur:

- Ross, Hans-Leo: Funktionale Sicherheit im Automobil, Hanser Verlag
- Löw, Pabst, Petry: Funktionale Sicherheit in der Praxis, dpunkt.Verlag

IFM5601 Deep Learning

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Jürgen Brauer
Dozent(en):	Prof. Dr. Jürgen Brauer
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Analysis, Lineare Algebra, Programmierkenntnisse
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden kennen mindestens drei der wichtigsten DL Architekturen im Detail
- Für jeden dieser drei Ansätze haben Sie fundiertes Wissen über den Aufbau und Funktionsweise gesammelt und können dieses Wissen bei Problemstellungen im Bereich Bildverarbeitung und Spracherkennung durch Anwendung eines geeigneten DL Verfahrens umsetzen
- Die Studierenden verstehen, welche Eigenschaften die DL Verfahren besonders auszeichnet und von klassischen Machine Learning Verfahren unterscheidet
- Sie kennen die historische Entwicklung des DL Gebietes und können abschätzen welche weiteren Entwicklungen zu erwarten sind

Lehrinhalte:

Einführung

- Motivation Deep Learning: kürzliche Erfolge
- Geschichte des Deep Learnings
- Gründe für den Boom
- Das biologische Vorbild
- Technische Neuronenmodelle
- Einfache technische Neuronale Netze
- Perzeptron / Perzeptron-Lernregel

- Multi-Layer-Perzeptron / Backpropagation-Algorithmus
- Deep Learning Architekturen
- CNN: Convolutional Neural Networks für Bildverarbeitungsaufgaben
- LSTM: Long Short Term Memory zur Automatischen Spracherkennung und Sprachübersetzung
- GAN: Generative Adversarial Networks
- Neural Style Transfer
- Deep Learning Tricks
- Ausblick: wohin entwickelt sich der Ansatz?

Literatur:

- Yann LeCun, Yoshua Bengio, Geoffrey Hinton. Deep Learning. Nature, 2015.
- Jürgen Schmidhuber. Deep Learning in Neural Networks: An Overview. Neural Networks Journal 61 (2015): 85-117
- Softwarebibliothek TensorFlow: <https://www.tensorflow.org/>



IFM5602 Mustererkennung und maschinelles Lernen

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Rieck
Dozent(en):	Prof. Dr. Stefan Rieck
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung/Praktikum
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden beherrschen die grundlegenden Begriffe des Maschinellen Lernens (ML). Sie können Verfahren der Mustererkennung und des Maschinellen Lernens adäquat auswählen und Mustererkennungsaufgaben in der Praxis lösen.

Lehrinhalte:

- Skripting in Python
- Grundlagen des Maschinellen Lernens
- Data Preprocessing (Vorverarbeitung, Merkmale)
- Grundlegende Verfahren zur Erkennung einfacher Muster (Bayes Classifier, Support Vector Machines, Neuronale Netze)
- Erkennung komplexer Musterfolgen und strukturierter Muster
- Kombination von Klassifikatoren

Literatur:

- Christopher M. Bishop: Pattern Recognition and Machine Learning. Springer, Berlin 2006, ISBN 387310738.
- Anselm Lingnau: Programmieren in Python: Eine praktische Einführung, online: <https://www.tuxcademy.org/download/de/pyth/pyth-de-manual.pdf>
- Pedregosa et al.: Scikit-learn: Machine Learning in Python, JMLR 12, pp. 2825-2830, 2011.
- Aurélien Géron. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques for Building Intelligent Systems. O'Reilly. 2017, ISBN:

1491962291

- Sebastian Raschka: Machine Learning mit Python: Das Praxis-Handbuch für Data Science, Predictive Analytics und Deep Learning, mitp 2016, ISBN: 3958454224



IFM5603 Simulation und Datenvisualisierung

Allgemeines

Die Lehrveranstaltung vermittelt Techniken zur Implementierung der Simulation von Prozessen und physikalischen Phänomenen sowie Methoden der wissenschaftlichen Visualisierung zur intuitiven Darstellung und visuellen Analyse von großen Datenmengen.

Modulverantwortliche(r):	Prof. Dr. Christoph Bichlmeier
Dozent(en):	Prof. Dr. Christoph Bichlmeier
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Grundlegende Kenntnisse zu der OpenGL-Rendering Pipeline und der Computergraphik sowie in der C++ Programmierung
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen (verpflichtend)
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übungen/Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Erfolgreiche Teilnahme an den Übungen, schriftl. Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme an der Lehrveranstaltung kennen die Studierenden verschiedene Anwendungsbereiche der Simulation und Visualisierung. Sie können für einige diese Anwendungsbereiche selbständig die erlernten Algorithmen implementieren.

Lehrinhalte:

Die Studierenden können mithilfe von Excel, C++, Qt, OpenGL und GLSL Lösungen für folgende Themenbereiche realisieren.</p>

- Wissenschaftliche Durchführung einer Datenerhebung und Datenauswertung
- Visuelle Darstellung von statistischen Datenerhebungen
- Visualisierung von Volumendaten und medizinischen Schnittbildern
- Visualisierung von Strömungsdaten
- Simulation von Weichgewebe
- Implementierung und Visualisierung von KI-Algorithmen zur Simulation zur Nachbildung von menschlichem Verhalten

Literatur:

- Artificial Intelligence for Games; 18. September 2009 von Ian Millington , John Fun-

ge

- Visual Computing for Medicine (Second Edition)□Theory, Algorithms, and Applications; Bernhard Preim and Charl P. Botha
- Visualisierung: Grundlagen und allgemeine Methoden; 2013; Heidrun Schumann, Wolfgang Müller
- Diverse Paper aus verschiedenen Quellen (Fachzeitschriften und Konferenzbände)



IFM5604 Parallelprogrammierung

Allgemeines

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

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden kennen aktuelle parallele Rechnerarchitekturen einordnen und deren Vor- und Nachteile
- Sie verstehen die wichtigsten parallele Programmierkonzepte
- Sie können die Leistung von Parallelrechnern bei der Lösung praktischer Probleme bewerten
- Sie kennen die Vorgehensweisen bei der effektiven Parallelisierung von Algorithmen
- Sie beherrschen den effektiven Einsatz moderner paralleler Programmierwerkzeuge

Lehrinhalte:

- Einführung
- Leistungsbewertung, Amdahl und Gustafson-Gesetz
- parallele Rechnerarchitekturen
- Programmierkonzepte für Parallelrechner
- OpenMP und MPI
- GPU-Programmierung mit CUDA und OpenCL
- PRAM (Parallel Random Access Machine)
- Parallele Algorithmen und Komplexität
- Präfixberechnungen
- Parallele Berechnung numerischer Ausdrücke

- Parallele Matrix- und Sortieralgorithmen
- Grid- und Cloud-Computing
- Quantenrechner

Literatur:

- Tanenbaum, A.S.: "Computerarchitektur. Strukturen - Konzepte - Grundlagen", Pearson Studium, 2006.
- "OpenMP Application Program Interface", Version 4.0 - RC 1, November 2012.
- "MPI: A Message-Passing Interface Standard", Version 3.0, Message Passing Interface Forum, September 21, 2012.
- "CUDA C PROGRAMMING GUIDE", NVIDIA, 2012.
- "Introduction to OpenCL Programming", AMD, 2010.
- Homeister, M.: "Quantum Computing verstehen", Springer Fachmedien Wiesbaden, 2015.
- Balaji, P.: "Compute Unified Device Architecture," Programming Models for Parallel Computing , MITP, 2015.
- Rauber, Th.; Runger, G.: "Parallele Programmierung", Springer Vieweg Berlin Heidelberg, 2012.
- Rahm, E.; Sattler, K.-U.: " Verteiltes und Paralleles Datenmanagement", Springer Vieweg Berlin Heidelberg, 2015.

IFM5605 Algorithmen und Strategien zur Entscheidungsunterstützung

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Jochen Staudacher
Dozent(en):	Prof. Dr. Jochen Staudacher
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit Vorlesung 15 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Die erfolgreiche Teilnahme an den Übungen gilt als Zulassungsvoraussetzung für eine 20-minütige mündliche Prüfung am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden können entscheidungs- und spieltheoretische Konzepte auf praktische Fragestellungen aus Informatik und Wirtschaftswissenschaften anwenden.
- Sie können beschreiben und analysieren, welchen Einfluss die Struktur eines sozialen oder ökonomischen Netzwerks auf strategisches Handeln besitzt.
- Sie können für ausgewählte Anwendungsfälle aus großen Datensätzen sinnvolle Information extrahieren, die zur Entscheidungsunterstützung eingesetzt werden kann.
- Sie können die vorgestellten Methoden und Algorithmen in R umsetzen.
- Sie können zu den vorgestellten Ansätzen und Algorithmen zur Entscheidungsunterstützung kritisch Stellung nehmen und deren Potenziale und Grenzen charakterisieren.

Lehrinhalte:

- Entscheidungskriterien, Entscheidungsmodelle, Entscheidungsalgorithmen
- Nutzenfunktionen, rationales Entscheiden bei Risiko, Bernoulli-Prinzip
- Interaktives Entscheiden, Algorithmische Spieltheorie und Anwendungen in Informatik und Ökonomie
- Anpassung und Erweiterung spieltheoretischer Ansätze zur Analyse ökonomischer und sozialer Netzwerke

- Ökonomien mit Netzwerkeffekten
- Statistische Entscheidungstheorie
- Datenbasiertes Entscheiden: Algorithmen und Anforderungen an die Datenqualität

Literatur:

Es gibt nicht das EINE Lehrbuch zu dieser Vorlesung; verschiedene Teile der Vorlesung werden durch die folgenden Lehrbücher abgedeckt:

- James N. Webb: Game Theory: Decisions, Interaction and Evolution, Springer, 2007
- Michael Maschler, Eilon Solan, Shmuel Zamir: Game Theory, Cambridge University Press, 2013
- Hans-Jürgen Zimmermann: Operations Research, Vieweg, 2. Auflage, 2008
- Matthew O. Jackson: Social and Economic Networks, Princeton University Press, 2008
- David Easley, Jon Kleinberg: Networks, Crowds, and Markets: Reasoning About a Highly Connected World, Cambridge University Press, 2010
- Pierre Lafaye de Micheaux, Remy Drouilhet, Benoit Liquet: The R Software, Springer, 2014
- Larry Wasserman: All of statistics, Springer, 2004



IFM5606 Big Data

Allgemeines

Die Datenmengen, die in unserer Kommunikations- und Wissensgesellschaft anfallen wachsen unaufhörlich und die Datenvielfalt nimmt zu. Sie bieten ein riesiges Potential um mehr über uns, unsere Gesellschaft, unsere Umwelt und unsere Wirtschaft zu erfahren. Daten müssen jedoch zunächst einmal gespeichert, verarbeitet und analysiert werden. Dies stellt ganz neue Anforderungen an die zugrundeliegenden Systeme, da die klassischen Ansätze mit relationalen Datenbanken für strukturierte Daten hierfür nicht mehr geeignet sind. In Big Data Systemen werden die Daten verteilt auf viele Knoten gespeichert und verarbeitet. Die Daten selbst sind oft semistrukturiert oder unstrukturiert und müssen ggf. erst aufbereitet und gefiltert werden. Analysemethoden werden immer ausgefeilter und zunehmend ist eine Analyse in Echtzeit gefordert, die wiederum neue Technologien und Systeme erfordert.

In der Vorlesung zu Big Data werden verschiedene Problemstellungen und Lösungsansätze für Big Data Systeme diskutiert. Im Praktikum werden diese Ansätze mit einigen der derzeit leistungsfähigsten Systeme auf reale Daten angewendet.

Modulverantwortliche(r):	Prof. Nikolaus Steger
Dozent(en):	Prof. Nikolaus Steger
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Praktikum 90 Selbststudium
Leistungsnachweis und Prüfung:	Aktive Teilnahme am Praktikum. Schriftliche Prüfung 90 Minuten am Ende des Semesters. Erfolgreiche Teilnahme am Praktikum ist Zulassungsvoraussetzung für die Prüfung.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage

- die Funktionsweise, die theoretischen Grundlagen und wichtige Algorithmen und Datenstrukturen zur Speicherung, Analyse und verteilten Verarbeitung großer Datenmengen zu verstehen
- konkrete Komponenten für Big Data Systeme zu kennen und in der Praxis einzusetzen
- komplexe Big Data Systeme zu konzipieren und aufzubauen



Lehrinhalte:

- Einsatzbereiche von Big Data Systemen
- Map-Reduce-Paradigma, Apache Hadoop, HDFS
- Batchverarbeitung mit Apache Spark, Resilient Distributed Datasets und Spark SQL
- Data Warehousing
- SQL-Erweiterungen für Analytics und Data Warehousing
- Column Store, Datenkomprimierung und Datenstrukturen zur Speicherung analytischer Daten
- Online Analytics und Stream Processing mit Apache Flink und Flink SQL
- Anforderungen, allgemeine Konzepte und Komponenten für Big Data Systeme

Literatur:

- Learning Spark, Lightning-Fast Data Analysis- Holden Karau, Andy Konwinski, O'Reilly and Associates
- Advanced Analytics with Spark: Patterns for Learning from Data at Scale - Sandy Ryza, Uri Laserson, Sean Owen, Josh Wills, O'Reilly and Associates



IFM5607 Algorithmen für Real Time Rendering

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Bernd Dreier
Dozent(en):	Prof. Dr. Bernd Dreier
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Grundlegende Kenntnisse der OpenGL-Rendering Pipeline entsprechend dem im Bachelor Informatik - Game Engineering angebotenen Fach
Verwendbarkeit:	Studiengang Informatik (M.Sc.), Studiengang Game Engineering und Visual Computing (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS betreutes Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweise im Praktikum, Leistungsnachweise sind Zulassungsvoraussetzung schriftl. Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden haben ein vertieftes Verständnis der Rendering-Pipeline
- Die Studierenden verstehen die Anforderungen neuerer OpenGL-Versionen (3.x, 4.x) und können dafür komplexe Shader auf der Basis von GLSL entwickeln
- Die Studierenden können komplexe Echtzeit-Anwendungen mit OpenGL realisieren
- Die Studierenden kennen Vor- und Nachteile verschiedener Arten der Speicherung polygonaler Netze und können diese auswählen und implementieren
- Die Studierenden verstehen die mathematischen Hintergründe moderner Grafikanwendungen
- Die Studierenden verstehen Interpolationsverfahren wie z.B. NURBS und können diese in der Computergrafik und Games anwenden
- Die Studierenden kennen Vulkan und Vulkan Raytracing und können Anwendungen mittlerer Komplexität damit entwickeln
- Die Studierenden kennen ausgewählte Algorithmen der algorithmischen Geometrie und deren Anwendung für Computergrafik und Games

Lehrinhalte:

- Shaderprogrammierung mit GLSL (Stand OpenGL 4.1)
- OpenGL Puffer und Off-Screen-Rendering

- Bump-/Shadowmapping, TBN-Koordinatensystem
- Perspektivisch korrekte Interpolation
- Speicherung und algorithmische Behandlung polygonaler Netze
- Bézier-Kurven, Splines und NURBS
- Neue Konzepte in OpenGL 3/4
- Vulkan und Vulkan Raytracing
- Bounding Volumes und ausgewählte Algorithmen der algorithmischen Geometrie

Literatur:

- Eric Lengyel, Mathematics for 3D Game Programming and Computer Graphics, 3rd Edition, 2011
- Edward Angel, Interactive Computer Graphics: A Top-Down Approach with Shader-Based OpenGL, Addison-Wesley, 2011
- Akenine-Möller, Real-Time Rendering, AK Peters, 2008
- Computational Geometry, Mark de Berg, et al., Springer, 2008



IFM5607 E-Business Management

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Wind
Dozent(en):	Prof. Dr. Stefan Wind
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung/Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden

- sind in der Lage, die durch elektronische Technologien induzierten Veränderungen traditioneller Geschäftsprozesse zu erklären
- sind durch ein grundsätzliches Verständnis des E-Business befähigt, die Möglichkeiten innovativer Verfahren zur Information, Kommunikation und Transaktion zu beschreiben
- kennen elektronische Geschäftsprozesse und -modelle in der Net Economy
- können diese Kenntnisse auf elektronische Kontaktnetzwerke (E-Community), den elektronischen Handel (E-Marketplace), Systeme bei der elektronischen Kooperation (E-Company), Systeme im elektronischen Einkauf (E-Procurement) und Systeme im elektronischen Verkauf (E-Shop) transferieren

Lehrinhalte:

- Grundlagen des E-Business Managements
- Geschäftsprozessmanagement im E-Business
- Prozesse, Systeme, Management, Integration und Implementierungen in den Bereichen E-Procurement, E-Community, E-Company, E-Shop und E-Marketplace
- Neuen Trends und Technologien im E-Busienss

Literatur:

- Ferstl O. K.: Sinz E.: Grundlagen der Wirtschaftsinformatik. 6. Aufl., Oldenbourg, München 2008
- Kollmann, T.: E-Business - Grundlagen elektronischer Geschäftsprozesse in der Net

Economy. 5. Auflage. Gabler, Wiesbaden 2013

- Meier, A. und Stormer H.: eBusiness & eCommerce : Management der digitalen Wertschöpfungskette. 2. Aufl., Springer Verlag, Berlin 200
- Wirtz B. W.: Electronic Business. 2 Aufl. Gabler, Wiesbaden 2001



IFM5100 Hauptseminar

Allgemeines

Modulverantwortliche(r):	Alle Professoren der Fakultät
Dozent(en):	Alle Professoren der Fakultät
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	Seminar
Leistungspunkte:	5
Arbeitsaufwand:	15 Stunden Präsenzzeit Vorträge und anschließende Diskussion 135 Stunden Vorbereitung der Präsentation und Ausarbeitung der Studienarbeit
Leistungsnachweis und Prüfung:	Seminarvortrag mit anschließender Diskussion Studienarbeit

Zur Prüfung zugelassene Hilfsmittel:

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage

- sich in ein anspruchsvolles, wissenschaftliches Thema einzuarbeiten
- entsprechende Literatur zu recherchieren und aufzuarbeiten
- einen Vortrag zu einem wissenschaftlichen Thema zu erarbeiten, zu präsentieren und bei der nachfolgenden Diskussion zu vertreten
- eine schriftliche Ausarbeitung zu einem wissenschaftlichen Thema nach den Grundsätzen guter wissenschaftlicher Praxis zu erstellen

Lehrinhalte:

Abhängig vom Thema des Hauptseminars

Literatur:

Abhängig vom Thema des Hauptseminars in Absprache mit dem Dozenten

- Balzert, H., Schröder, M., Schäfer, C.: Wissenschaftliches Arbeiten, W3L Verlag, 2. Auflage, 2012
- Wytrzens, Hans Karl: Wissenschaftliches Arbeiten : eine Einführung, 5., überarbeitete Auflage, Wien, ISBN: 9783708915005
- DFG: Sicherung guter wissenschaftlicher Praxis - Denkschrift, WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim, ergänzte Auflage, 2013



IFM5200 Projekt

Allgemeines

Modulverantwortliche(r):	Alle Professoren der Fakultät
Dozent(en):	Alle Professoren der Fakultät
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	Projekt
Leistungspunkte:	5
Arbeitsaufwand:	15 Stunden Präsenzzeit Besprechung mit betreuendem Professor 135 Stunden Durchführung des Projektes
Leistungsnachweis und Prüfung:	Bewertung des Projektergebnisses durch betreuenden Professor

Zur Prüfung zugelassene Hilfsmittel:

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage

- ein Projekt zu planen, zu organisieren und zu leiten
- ein Projekt zu dokumentieren und gegenüber Dritten zu vertreten
- Probleme im Projekt frühzeitig zu erkennen und zu lösen
- ein Projekt erfolgreich abzuschließen
- das erworbene Wissen auch in komplexen Anwendungsszenarien einzusetzen und erfolgreich anzuwenden
- aus unvollständigen und ggf. widersprüchlichen Anforderungen in Abstimmung mit dem Auftraggeber konsistente und überzeugende Lösungen zu erarbeiten
- sich schnell in neue Anwendungsgebiete, Technologien und Grundlagen einzuarbeiten

Lehrinhalte:

Abhängig vom Projektauftrag in Absprache mit dem jeweiligen Dozenten

Literatur:

Abhängig vom Projektauftrag in Absprache mit dem jeweiligen Dozenten

IFM6101 Masterarbeit

Allgemeines

Modulverantwortliche(r):	Betreuender Professor
Dozent(en):	Betreuender Professor
Modultyp:	Pflichtmodul
Voraussetzungen:	mindestens 35 CP und in mindestens 4 Pflichtmodulen die Note
Verwendbarkeit:	Studiengang Informatik (M.Sc.)
Angebot und Dauer:	Die Frist zur Bearbeitung der Masterarbeit im Vollzeitstudium beträgt 6 Monate, im Teilzeitstudium 12 Monate.
Lehrformen:	
Leistungspunkte:	30
Arbeitsaufwand:	900 Stunden
Leistungsnachweis und Prüfung:	Abschlussarbeit
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

- Fähigkeit die im Studium erworbenen Kenntnisse und Fertigkeiten in einer selbständig angefertigten, anwendungsorientiert-wissenschaftlichen Arbeit auf komplexe Aufgabenstellungen anzuwenden
- Fähigkeit sich selbständig in ein wissenschaftliches Thema einzuarbeiten
- Fähigkeit aus unvollständigen und widersprüchlichen Informationen in Abstimmung mit dem Aufgabensteller eine konsistente und überzeugende Lösung zu erarbeiten
- Die Abschlussarbeit darf mit Zustimmung der Prüfungskommission in einer Einrichtung außerhalb der Hochschule ausgeführt werden.

Lehrinhalte:

Literatur:

In Absprache mit dem betreuenden Professor

- Wytrzens, Hans Karl: Wissenschaftliches Arbeiten : eine Einführung, 5., überarbeitete Auflage, Wien, ISBN: 9783708915005
- DFG: Sicherung guter wissenschaftlicher Praxis - Denkschrift, WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim, ergänzte Auflage, 2013