

UNIVERSITÄT PADERBORN

FAKULTÄT FÜR ELEKTROTECHNIK, INFORMATIK UND MATHEMATIK
INSTITUT FÜR INFORMATIK

MODULHANDBUCH FÜR DEN
MASTERSTUDIENGANG COMPUTER SCIENCE v1 (CSMA v1)

STAND: 10. JULI 2026

Inhaltsverzeichnis

1	Präambel und Hinweise	3
2	Pflichtmodule	4
3	Wahlpflichtmodule	16
4	Focus Areas	156
4.1	Classical and Quantum Algorithm Design	156
4.2	Computer and Communication Systems	157
4.3	Data Science and Intelligent Systems	157
4.4	Security	159
4.5	Software Engineering	160
5	Module im Wintersemester	161
6	Module im Sommersemester	162

1 Präambel und Hinweise

Aus technischen Gründen wurde die Präambel des Modulhandbuches ausgelagert. Sie ist unter Prüfungsangelegenheiten auf den Seiten zum Studium des Instituts für Informatik zu finden. Wir bitten um Beachtung dieser Präambel. Bei Fragen zu dieser Präambel wenden Sie sich bitte an das Studienbüro Informatik.

Bitte beachten Sie auch, dass

1. in diesem Modulhandbuch alle laut Prüfungsordnung vorgesehenen Module aufgelistet werden, auch wenn sie in dem entsprechenden Semester nicht angeboten werden.
2. dieses Modulhandbuch den Datenbestand zum Zeitpunkt der Erstellung beinhaltet.

2 Pflichtmodule

Master Thesis							
Master Thesis							
Modulnummer: A.079.4010		Workload (h): 900	Leistungspunkte: 30		Turnus: Sommer- / Wintersemester		
		Studiensemester: 4	Dauer (in Sem.): 1		Sprache: en		
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	2024.7090a Master Thesis – Work Plan		30	120	P	1
	b)	2024.7090b Master Thesis		30	720	P	1
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: Modulprüfungen im Umfang von 48 LP müssen erfolgreich abgelegt worden sein. Wird eine Spezialisierung angestrebt, müssen mindestens drei Module in der Spezialisierung erfolgreich abgeschlossen sein und das Thema der Masterarbeit im Rahmen der Spezialisierung gewählt werden.						

4	Inhalte: <i>Inhalte der Lehrveranstaltung Master Thesis – Work Plan:</i> Nach Themenabsprache mit dem Betreuer bzw. der Betreuerin erfolgt eine erste grobe Einarbeitung. Auf dieser Grundlage und einer ersten Literaturrecherche ist durch den Studenten bzw. die Studentin ein Arbeitsplan vorzulegen, der die zu erzielenden Ergebnisse samt Meilensteinen für die Arbeit dokumentiert. Die Beschreibung der zu bearbeitenden Aufgabe und Zielsetzung soll einen Umfang von in der Regel nicht mehr als fünf (in Ausnahmen bis zu zehn) Seiten haben und nach Ausgabe des Themas innerhalb eines Monats beim Betreuer bzw. der Betreuerin eingereicht werden. Der Antrittsvortrag soll zumindest die geplante Arbeit, das geplante Vorgehen und die erhofften Ergebnisse vorstellen und eine Dauer von ca. 30 Minuten haben. An den Antrittsvortrag schließt sich eine Aussprache an. <i>Inhalte der Lehrveranstaltung Master Thesis:</i> Die Masterarbeit umfasst die Bearbeitung eines Themas mit schriftlicher Ausarbeitung und einer mündlicher Präsentation der Ergebnisse. In der Masterarbeit zeigt der bzw. die Studierende seine bzw. ihre Fähigkeit zum selbständigen wissenschaftlichen Arbeiten an einem angemessen anspruchsvollen Thema, das auch Gelegenheit zur Entfaltung eigener Ideen gibt. Auf der Grundlage des “State-of-the-art” sollen die Methoden der Informatik systematisch angewendet werden. Die Aufgabe einer Masterarbeit kann beispielsweise die Entwicklung von Software, Hardware, eine Beweisführung oder eine Literaturrecherche umfassen. Die Masterarbeit soll einen Umfang von 120 Seiten nicht überschreiten. Die Ergebnisse der Masterarbeit sind in einer fakultätsöffentlichen Abschlusspräsentation mit anschließender wissenschaftlicher Aussprache vorzustellen. Diese findet nach Abgabe der schriftlichen Arbeit statt (in der Regel spätestens nach vier Wochen) und soll eine Dauer von ca. 30 Minuten haben. Die Abschlusspräsentation kann neben Vortrag und Aussprache weitere Elemente (z.B. die Demonstration von Software) beinhalten. Die Abschlusspräsentation geht in die Endnote mit ein. Konkrete Aufgabenstellungen für Masterarbeiten werden laufend auf den Webseiten der Fachgebiete des Instituts für Informatik veröffentlicht.
---	---

2 Pflichtmodule

5	Lernergebnisse und Kompetenzen: Im Rahmen ihrer Masterarbeit bearbeiten die Studierenden ein Problem nach wissenschaftlichen Methoden innerhalb einer bestimmten Frist. Die im Zuge des Studiums erworbenen fachlich-methodischen sowie fachübergreifenden Kompetenzen sollen dazu entsprechend eingesetzt werden. Dazu gehören insbesondere auch die Strukturierung und Planung der einzelnen Arbeitsschritte sowie die Präsentation der Ergebnisse nach Abschluss der Arbeit. Durch die Masterarbeit wird nicht nur das Fachwissen in einem spezifischen Themengebiet der Informatik vertieft, es werden vor allem auch wichtige "Soft Skills" eingeübt, die für die spätere berufliche Praxis essentiell sind. Die Studierenden können • ein Problem aus dem Bereich der Informatik innerhalb einer bestimmten Frist eigenständig unter fachlicher und methodischer Betreuung mit wissenschaftlichen Methoden bearbeiten, • die im Zuge des Studiums erworbenen fachlich-methodischen fachübergreifenden Kompetenzen zur Lösung komplexer Aufgabenstellungen anwenden, • die selbständige Bearbeitung einer umfangreichen Aufgabenstellung planen und durchführen, • Anforderungen und Rahmenbedingungen einer umfangreichen Aufgabenstellung mit Betreuenden klären, • eigene Kreativität zur Problemlösung einsetzen, • selbständig und zielgerichtet Fachwissen und Methoden erarbeiten, um Teilprobleme zu lösen, • die Ergebnisse in Form einer wissenschaftlichen Arbeit schriftlich und mündlich präsentieren.														
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)														
	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">zu</th><th style="width: 50%;">Prüfungsform</th><th style="width: 20%;">Dauer bzw. Umfang</th><th style="width: 20%;">Gewichtung für die Modulnote</th></tr> </thead> <tbody> <tr> <td>a) - b)</td><td>Masterarbeit mit Abschlusspräsentation</td><td>30-120 Seiten bzw. ca. 30 min</td><td>100%</td></tr> </tbody> </table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a) - b)	Masterarbeit mit Abschlusspräsentation	30-120 Seiten bzw. ca. 30 min	100%						
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote												
a) - b)	Masterarbeit mit Abschlusspräsentation	30-120 Seiten bzw. ca. 30 min	100%												
7	Studienleistung, qualifizierte Teilnahme: <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">zu</th><th style="width: 50%;">Form</th><th style="width: 20%;">Dauer bzw. Umfang</th><th style="width: 20%;">SL / QT</th></tr> </thead> <tbody> <tr> <td>a)</td><td>Arbeitsplan und Antrittsvortrag</td><td>bis zu 5 Seiten bzw. 30 min</td><td>QT</td></tr> <tr> <td>b)</td><td>keine</td><td></td><td></td></tr> </tbody> </table> Vom jeweiligen Lehrenden wird spätestens bei der Ausgabe des Themas bekannt gegeben, wie die qualifizierte Teilnahme konkret zu erbringen ist.			zu	Form	Dauer bzw. Umfang	SL / QT	a)	Arbeitsplan und Antrittsvortrag	bis zu 5 Seiten bzw. 30 min	QT	b)	keine		
zu	Form	Dauer bzw. Umfang	SL / QT												
a)	Arbeitsplan und Antrittsvortrag	bis zu 5 Seiten bzw. 30 min	QT												
b)	keine														
8	Voraussetzungen für die Teilnahme an Prüfungen: Der Nachweis der Qualifizierten Teilnahme ist Voraussetzung für die Modulabschlussprüfung.														
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Masterarbeit einschließlich des Abschlussvortrages mit fachlicher Diskussion bestanden wurde.														

2 Pflichtmodule

10	Gewichtung für Gesamtnote: Das Modul wird mit 50 Gewichtspunkten gewichtet (Faktor 5/3).
11	Verwendung des Moduls in anderen Studiengängen: keine
12	Modulbeauftragte/r: Prof. Dr.-Ing. Juraj Somorovsky
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Master Thesis – Work Plan:</i> Methodische Umsetzung Direkte Absprache mit Betreuer. Lernmaterialien, Literaturangaben Je nach gewähltem Thema. <i>Hinweise der Lehrveranstaltung Master Thesis:</i> Methodische Umsetzung Selbständiges Arbeiten unterstützt durch individuelle Betreuung Lernmaterialien, Literaturangaben Je nach gewähltem Thema.

2 Pflichtmodule

Project Group							
Project Group							
Modulnummer: M.079.4201		Workload (h): 600		Leistungspunkte: 20		Turnus: Sommer- / Wintersemester	
		Studiensemester: 2-3		Dauer (in Sem.): 2		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	2024.7091 Project Group	PG	240	360	P	16
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine Teilnahmevoraussetzungen der Lehrveranstaltung Project Group: Empfohlene Vorkenntnisse Abhängig vom Thema.						
4	Inhalte: Inhalte der Lehrveranstaltung Project Group: In einer Projektgruppe arbeitet eine Gruppe von in der Regel 8-16 Studierenden über einen Zeitraum von einem Jahr (zwei Semester) gemeinsam an einem vom Gruppenorganisator festgelegten Forschungsthema. Projektgruppen führen die Studierenden in aktuelle Forschungsthemen ein, die in der Regel mit dem speziellen Interessengebiet des Gruppenleiters zusammenhängen; die Teamarbeit der Projektgruppe soll eine Vorbereitung auf die industrielle Praxis sein. Die Themen der Projektgruppen decken das gesamte Spektrum der Forschungsinteressen der Forschungsgruppen des Fachbereichs Informatik ab. In der Regel werden die Projektgruppenmitglieder in Unterteams aufgeteilt. Das Team wählt einen Projektgruppenleiter, der die Arbeit des gesamten Teams steuert. Der Leiter ist für die interne Kommunikation und die Meldung möglicher Probleme an den Organisator der Gruppe verantwortlich. Die Teammitglieder treffen sich regelmäßig, um ihre Fortschritte mit dem Gruppenorganisator zu besprechen und über den aktuellen Stand ihrer Arbeit zu berichten. Die Teammitglieder bestehen dieses Modul erfolgreich, wenn sie den Quellcode und den Gruppenbericht einreichen und eine Abschlusspräsentation halten. Weitere Auflagen können vom Gruppenorganisator festgelegt werden.						

2 Pflichtmodule

5	Lernergebnisse und Kompetenzen: In der Projektgruppe wird Teamarbeit und Organisation eines Projekts praktisch erprobt und erlernt; hierdurch werden die Teilnehmenden auf die spätere industrielle Berufspraxis vorbereitet. Die Studierenden lernen umfangreiche Entwicklungsprozesse im Team aus eigener Anschauung kennen. Durch die ausdrückliche Arbeitsteilung entsteht der Zwang, über eigene Arbeiten innerhalb der Gruppe zu berichten und die Ergebnisse zu vertreten. Nichtkognitive Kompetenzen • Einsatz und Engagement • Gruppenarbeit • Lernkompetenz • Lernmotivation • Motivationale und volitionale Fähigkeiten • Schreib- und Lesekompetenz (wissenschaftlich) • Selbststeuerungskompetenz										
6	Prüfungsleistung: ☐ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☒ Modulteilprüfungen (MTP)										
	<table border="1"> <thead> <tr> <th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr> </thead> <tbody> <tr> <td>a)</td><td>Software mit Dokumentation, Referat</td><td></td><td>75%, 25%</td></tr> </tbody> </table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Software mit Dokumentation, Referat		75%, 25%		
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote								
a)	Software mit Dokumentation, Referat		75%, 25%								
	Im Modul Projektgruppe sind regelmäßig Berichte über den Arbeitsfortschritt zu erstellen. Außerdem ist die erfolgreiche Bearbeitung von Projekten durch die Abgabe von Software und Dokumentation nachzuweisen. Die Ergebnisse der Projektarbeiten sind in einer Präsentation vorzustellen. Es wird eine Note für die Gesamtheit der Teilleistungen vergeben. Die Softwareprojekte mit Dokumentation bilden 75% der Modulnote, das Referat bildet 25% der Modulnote.										
7	Studienleistung, qualifizierte Teilnahme:										
	<table border="1"> <thead> <tr> <th>zu</th><th>Form</th><th>Dauer bzw. Umfang</th><th>SL / QT</th></tr> </thead> <tbody> <tr> <td>a)</td><td>Fortschrittsberichte oder Referate</td><td></td><td>SL</td></tr> </tbody> </table>	zu	Form	Dauer bzw. Umfang	SL / QT	a)	Fortschrittsberichte oder Referate		SL		
zu	Form	Dauer bzw. Umfang	SL / QT								
a)	Fortschrittsberichte oder Referate		SL								
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung										
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.										
10	Gewichtung für Gesamtnote: Das Modul wird mit 10 Gewichtspunkten gewichtet (Faktor 0,5).										
11	Verwendung des Moduls in anderen Studiengängen: keine										
12	Modulbeauftragte/r: Prof. Dr.-Ing. Juraj Somorovsky										

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Project Group:</i> Methodische Umsetzung • Es finden Plenumstreffen (alle Teilnehmenden und der Veranstaltenden) statt, insbesondere zur Vermittlung gemeinsam erforderlichen Wissens (Seminarphasen zu Beginn der beiden Semester) und zur Planung der weiteren Arbeit. • Es werden feste Verantwortlichkeiten zwischen den Teilnehmenden aufgeteilt, die über die gesamte Projektlaufzeit oder auch nur kurzfristig (ad-hoc-Aufgaben) Bestand haben können. • Es werden Untergruppen zu einzelnen Themen gebildet, die selbständig und termingebunden Aufgaben vorantreiben und dem Plenum Rechenschaft ablegen müssen. • Typischerweise erarbeitet jede Projektgruppe auch eine Repräsentation ihrer Arbeit in einer Webseite. • Am Ende jedes der beiden Semester ist ein Bericht zu erstellen, der in jedem Aspekt von den Teilnehmenden gestaltet und mit Inhalt gefüllt wird. Lernmaterialien, Literaturangaben Abhängig vom Thema.
----	---

2 Pflichtmodule

Key Skills							
Key Skills							
Modulnummer: M.079.4202		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommer- / Wintersemester	
		Studiensemester: 2		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehr- form	Kontakt- zeit (h)	Selbst- studium (h)	Status (P/WP)	Gruppen- größe (TN)
	a)	2024.7092 Wissenschaftliches Arbeiten	V1	15	15	P	150
	b)	2024.7092b Seminar	S2	30	120	P	15
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Seminar:</i> Empfohlene Vorkenntnisse Abhängig vom Seminarthema.						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Wissenschaftliches Arbeiten:</i> Die Inhalte des Moduls gliedern sich in vier Teile. Zunächst lernen die Studierenden den Umgang mit wissenschaftlicher Literatur; sie werden beispielsweise mit Methoden und Werkzeugen für die Suche nach verwandten Werken und die Verwaltung von Referenzen vertraut gemacht. Zweitens lernen die Studierenden, wie man wissenschaftliche Arbeiten verfasst. Dazu gehören typische Fehler und bewährte Verfahren zur Strukturierung von Arbeiten. Drittens lernen die Studierenden, wie sie wissenschaftliche Arbeiten begutachten und ihren Kommilitonen Feedback geben können, um ihre Arbeit zu verbessern. Schließlich lernen die Schülerinnen und Schüler, wie sie Präsentationen erstellen und halten können. Nach jedem Teil schreiben die Schüler einen kurzen Test. <i>Inhalte der Lehrveranstaltung Seminar:</i> In Seminaren erarbeiten sich die Teilnehmenden ein Thema, welches in einem Vortrag mit anschließender Diskussion und einer schriftlichen Ausarbeitung präsentiert wird. Vortragsmaterial und Ausarbeitung dienen dabei unterschiedlichen Zielen: Während das Vortragsmaterial zur Unterstützung des Vortrags dient (der in engen zeitlichen Grenzen abläuft), dient die Ausarbeitung dazu, sich zu einem späteren Zeitpunkt detailliert über das Thema informieren zu können. Ein Seminar beschäftigt sich in der Regel mit 8 bis 15 zusammenhängenden Teilthemen, die von je einem Teilnehmer bearbeitet werden. Die Seminarthemen decken das gesamte Spektrum der Forschungsgebiete der Fachgebiete des Instituts für Informatik ab.						

2 Pflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können • ein forschungsnahes Themengebiet der Informatik auf wissenschaftlichem Niveau selbstständig erarbeiten, insbesondere auch durch gründliche Literaturrecherche, die das Studium wissenschaftlicher Veröffentlichungen einschließt, • Konzepte und Sachverhalte erklären, die für das gewählte Themengebiet relevant sind, • geeignete Quellen auswählen und angemessen mit Literatur umgehen • Inhalte aus verschiedenen Quellen sichten, verstehen und zu einem Gesamtbild zusammenfügen, • erarbeitete Erkenntnisse in Form von Ausarbeitungen im wissenschaftlichen Stil und in Form einer wissenschaftlichen Präsentation darstellen und dabei inhaltliche Prioritäten setzen • einen Vortrag entlang einer inhaltlichen Linie strukturieren und verschiedene Mittel zur Illustration komplexer Sachverhalte nutzen, • fachliche Themen mit anderen diskutieren, • die eigenen Arbeitsweisen reflektieren und kommunizieren, • als Zuschauer aus einem Vortrag Kenntnisse aufnehmen und in Diskussionen Meinungen und Information austauschen.												
6	Prüfungsleistung: ☐ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☒ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td></td><td></td><td></td></tr><tr><td>b)</td><td>Seminarvortrag mit Diskussion, schriftliche Ausarbeitung</td><td>30-45 Minuten bzw. 15-30 Seiten</td><td>40%, 60%</td></tr></table> Der Seminarvortrag mit Diskussion bildet 40% der Modulnote, die schriftliche Ausarbeitung bildet 60% der Modulnote.	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)				b)	Seminarvortrag mit Diskussion, schriftliche Ausarbeitung	30-45 Minuten bzw. 15-30 Seiten	40%, 60%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote										
a)													
b)	Seminarvortrag mit Diskussion, schriftliche Ausarbeitung	30-45 Minuten bzw. 15-30 Seiten	40%, 60%										
7	Studienleistung, qualifizierte Teilnahme: <table><tr><th>zu</th><th>Form</th><th>Dauer bzw. Umfang</th><th>SL / QT</th></tr><tr><td>a)</td><td>Kurzklausur</td><td>max. 30 min</td><td>QT</td></tr><tr><td>b)</td><td>keine</td><td></td><td></td></tr></table>	zu	Form	Dauer bzw. Umfang	SL / QT	a)	Kurzklausur	max. 30 min	QT	b)	keine		
zu	Form	Dauer bzw. Umfang	SL / QT										
a)	Kurzklausur	max. 30 min	QT										
b)	keine												
8	Voraussetzungen für die Teilnahme an Prüfungen: keine												
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.												
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).												

2 Pflichtmodule

11	Verwendung des Moduls in anderen Studiengängen: keine
12	Modulbeauftragte/r: Prof. Dr.-Ing. Juraj Somorovsky
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Seminar:</i> Methodische Umsetzung Referate mit schriftlicher Ausarbeitung und Vortrag. Lernmaterialien, Literaturangaben Abhängig vom Seminarthema.

2 Pflichtmodule

General Studies							
General Studies							
Modulnummer:		Workload (h): 300	Leistungspunkte: 10		Turnus: Sommer- / Wintersemester		
		Studiensemester: 2-3	Dauer (in Sem.): 2		Sprache: en		
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	Studium Generale – Master	diverse	150	150	P	100
2	Wahlmöglichkeiten innerhalb des Moduls: Beliebige Veranstaltungen außerhalb der Informatik können gewählt werden.						
3	Teilnahmevoraussetzungen: keine						
4	Inhalte: Eine beliebige Kombination von Veranstaltungen außerhalb der Informatik im Umfang von maximal 10 LP muss gewählt werden. <i>Inhalte der Lehrveranstaltung Studium Generale – Master:</i> Abhängig von den gewählten Veranstaltungen						
5	Lernergebnisse und Kompetenzen: Die Studierenden erweitern ihren wissenschaftlichen Horizont über die Grenzen der Informatik hinaus. Je nach gewählter Veranstaltung haben sie Kompetenzen im Bereich Kommunikationsfähigkeit, Teamarbeit und Präsentationstechniken erworben. Die Studierenden • erwerben Wissen und Kompetenzen, die nicht fachspezifisch sind, aber für das angestrebte Berufsziel bedeutsam sein können wie beispielsweise spezielle Kenntnisse bei Fremdsprachen, in ingenieur-, natur-, kultur- oder wirtschaftswissenschaftlichen Gebieten, • analysieren unterschiedlichste Fragestellungen in den entsprechenden Gebieten, • ordnen das fachspezifische Wissen in einen interdisziplinären Zusammenhang ein, • stellen Zusammenhänge zum Informatik-Studium her, • erweitern ihre Schlüsselkompetenzen und ggf. Fremdsprachenkompetenzen, wodurch die Persönlichkeitsbildung unterstützt wird, auch in interkultureller Hinsicht.						
6	Prüfungsleistung:						

2 Pflichtmodule

7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Qualifizierte Teilnahme im Rahmen des Studium Generale	QT
8	Voraussetzungen für die Teilnahme an Prüfungen: keine		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die qualifizierte Teilnahme erbracht wurde.		
10	Gewichtung für Gesamtnote: Das Modul ist unbenotet.		
11	Verwendung des Moduls in anderen Studiengängen: keine		
12	Modulbeauftragte/r: Prof. Dr.-Ing. Juraj Somorovsky		
13	Sonstige Hinweise: keine		

3 Wahlpflichtmodule

Advanced Algorithms																					
Advanced Algorithms																					
Modulnummer: M.079.4002		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester															
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en															
1	Modulstruktur <table><tr><td></td><td>Lehrveranstaltung</td><td>Lehrform</td><td>Kontaktzeit (h)</td><td>Selbststudium (h)</td><td>Status (P/WP)</td><td>Gruppengröße (TN)</td></tr><tr><td>a)</td><td>L.079.05801 Advanced Algorithms</td><td>V3 Ü2</td><td>75</td><td>105</td><td>P</td><td>70/35</td></tr></table>								Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)	a)	L.079.05801 Advanced Algorithms	V3 Ü2	75	105	P	70/35
	Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)															
a)	L.079.05801 Advanced Algorithms	V3 Ü2	75	105	P	70/35															
2	Wahlmöglichkeiten innerhalb des Moduls: keine																				
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Advanced Algorithms:</i> Empfohlene Vorkenntnisse Bereitschaft und Fähigkeit, den kreativen Prozess des Algorithmenentwurfs und die Effizienzanalyse mit mathematischen Methoden zu erlernen. Grundkenntnisse einiger grundlegender Algorithmen und Datenstrukturen und deren Analysen werden vorausgesetzt.																				
4	Inhalte: <i>Inhalte der Lehrveranstaltung Advanced Algorithms:</i> Dieser Kurs präsentiert fortgeschrittene Algorithmen und algorithmische Paradigmen für grundlegende Probleme. Insbesondere werden dabei Methoden wie Randomisierung und Derandomisierung, sowie die Konzepte von Approximations- und Onlinealgorithmen anhand wichtiger algorithmischer Probleme vorgestellt. In allen Fällen werden Korrektheitsbeweise und Laufzeitanalysen durchgeführt. • Randomisierte Algorithmen und Derandomisierung, Beispiele u.a. Randomized Rounding • Online Algorithmen, Beispiele u.a. aus dem Bereich Scheduling • Approximationsalgorithmen, Beispiele u.a. NP-schwere Probleme																				

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können • grundlegende Analysetechniken verstehen und anwenden, • grundlegende algorithmische Ansätze erklären und anwenden, • beurteilen, welche Auswirkungen diese Ansätze haben und • die Grenzen der Anwendbarkeit dieser Ansätze einschätzen.			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch			
12	Modulbeauftragte/r: Prof. Dr. Christian Scheideler			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Advanced Algorithms:</i> Methodische Umsetzung Die Vorlesung nutzt Tafelanschrieb und Folien sowie kleine Aufgaben für die Studierenden während der Vorlesung. Sie wird durch Tutorien begleitet. Studierende haben in den Tutorien Gelegenheit, Aufgaben in der Gruppe zu bearbeiten und Lösungen von Übungsblättern mit den Tutoren zu diskutieren. Lernmaterialien, Literatur • Foliensatz der Vorlesung; Übungsblätter • Weitere Literatur wird in der Vorlesung bekanntgegeben
----	--

3 Wahlpflichtmodule

Advanced Complexity Theory							
Advanced Complexity Theory							
Modulnummer: M.079.4004		Workload (h): 180	Leistungspunkte: 6		Turnus: Sommersemester		
		Studiensemester: 1-3	Dauer (in Sem.): 1		Sprache: en		
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05803 Advanced Complexity Theory	V3 Ü2	75	105	P	25
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Advanced Complexity Theory:</i> Empfohlene Vorkenntnisse Grundlagen über Komplexitätstheorie (u.a. Turingmaschinen, NP-Vollständigkeit)						

3 Wahlpflichtmodule

4	Inhalte: <i>Inhalte der Lehrveranstaltung Advanced Complexity Theory:</i> Komplexitätstheorie beschäftigt sich mit der Bestimmung der Größe von Ressourcen (z.B. Laufzeit, Speicherverbrauch), die notwendig und hinreichend ist für die Lösung eines bestimmten algorithmischen Problems (z.B. Problem des Handlungsreisenden (TSP)) auf einem bestimmten Computermode (z.B. Turing-Maschine). Ein Ansatz ist die Komplexitätsklassen wie z.B. P, NP, PSPACE zu definieren, um die Problemkomplexität mit Hilfe der Vollständigkeit in einer solchen Klasse zu klassifizieren, wie z.B. die berühmte Klasse der NP-vollständigen Probleme. Dies ergibt bedingte Aussagen wie "Wenn NP nicht gleich P, dann ist TSP nicht in Polynomialzeit lösbar". Dieser Zweig der Komplexitätstheorie wird oft als strukturelle Komplexitätstheorie bezeichnet. Im Gegensatz dazu ist das Beweisen expliziter Untergrenzen für bestimmte Probleme das Thema der so genannten konkreten Komplexitätstheorie. Da niemand derzeit in der Lage ist superlineare Zeitschranken für explizit definierte Probleme in allgemeinen Rechenmodellen wie Turingmaschinen zu beweisen, betrachtet man etwas eingeschränkt Modelle wie 1-Band Turingmaschinen, monotone Bool'sche Schaltkreise, Bool'sche Schaltkreise mit beschränkter Tiefe, algebraische Berechnungsmodelle und verschiedene Arten von parallelen Berechnungsmodellen. Die Vorlesung gibt eine Übersicht von Ansätzen um solche unteren Schranke in verschiedenen Modellen zu beweisen. • Deterministische, nichtdeterministische und probabilistische Zeit- und Platz-Komplexitätsklassen, Hierarchien, Vollständigkeit • Untere Schranken für Größe und Tiefe verschiedener Varianten von von Booleschen Schaltkreisen • Untere Schranken für Algebraische Berechnungen								
5	Lernergebnisse und Kompetenzen: Die Studierenden können • verschiedene Berechnungsmodelle definieren und anwenden • wichtige Beweistechniken für untere Schranken beschreiben und erläutern • diese Beweistechniken auf neue Probleme anwenden • die Komplexität von Berechnungsproblemen einschätzen • geeignete Berechnungsmodelle identifizieren, um die Komplexität konkreter Probleme zu untersuchen								
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote						
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%						

3 Wahlpflichtmodule

7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: keine		
12	Modulbeauftragte/r: Prof. Dr. Johannes Blömer		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Advanced Complexity Theory:</i> Methodische Umsetzung Fließtext mit ca. 200-500 Zeichen Lernmaterialien, Literaturangaben • C.H. Papadimitriou, Computational Complexity, Addison-Wesley • S. Arora, B. Barak, Computational Complexity - A Modern Approach, Cambridge University Press • Foliensatz der Vorlesung, Übungsblätter		

3 Wahlpflichtmodule

Advanced Computer Architecture							
Advanced Computer Architecture							
Modulnummer: M.079.4005		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05724 Advanced Computer Architecture	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Advanced Computer Architecture:</i> Empfohlene Vorkenntnisse Kenntnisse aus der Bachelor-Veranstaltung Rechnerarchitektur sind hilfreich.						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Advanced Computer Architecture:</i> Die Lehrveranstaltung vermittelt die wesentliche Konzepte und Methoden, die beim Entwurf moderner Prozessoren Verwendung finden. Insbesondere werden dabei fortgeschrittene Aspekte der Optimierung von Zugriffszeiten und Durchsatz in der Speicherhierarchie, sowie Ansätze zur Nutzung von Parallelität auf der Instruktions-, Daten- und Thread-Ebene besprochen. Die Veranstaltung umfasst folgende Inhalte: • Grundlagen der Rechnerarchitektur (Wiederholung und Zusammenfassung) • Entwurf der Speicherhierarchie • Parallelität auf Instruktionsebene • Datenparallelität: Vektor-, SIMD- und GPU-Architekturen • Parallelität auf Thread-Ebene • Warehouse-scale Computer • Domain-specific computer architectures						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Nach Abschluss dieses Moduls können Studierende • die Architektur moderner Speichersysteme mit mehreren Ebenen erklären, die durchschnittliche Zugriffszeit mathematisch modellieren und den Einfluss der wesentlichen Entwurfsparameter qualitativ beschreiben und bewerten, • die Konzepte zur Parallelverarbeitung auf Daten-, Instruktions-, Thread-, und Task-Ebene erläutern und Algorithmen zur Out-of-Order Execution gegenüberstellen, • die Grenzen der Rechenleistung für konkrete Anwendungen und Architekturen anhand des des Roofline-Modells untersuchen, • die gebräuchlichen Ansätze und Protokolle für Cachekohärenz in Multiprozessor-Systemen erläutern und die Funktionsweise an Beispielen demonstrieren und • durch Computersimulation unterschiedliche Ausprägungen moderner Rechnersysteme quantitativ bewerten und die Resultate interpretieren.			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch			
12	Modulbeauftragte/r: Prof. Dr. Christian Plessl, Prof. Dr. Marco Platzner			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Advanced Computer Architecture:</i> Methodische Umsetzung Die Lehrveranstaltung besteht aus einer Vorlesung und Rechen- sowie praktischen Übungen. Die Vorlesung erfolgt mit Beamer und Tafelanschrieb. In den Rechenübungen werden Aufgaben ausgegeben und deren Lösungen in einer Übungseinheit vorgestellt und diskutiert. Im Rahmen der praktischen Übungen werden die Auswirkungen von Entwurfsentscheidungen und Optimierungsmöglichkeiten auf der Hard- und Softwareebene am Computer mit Simulatoren von Prozessor und Speichersystemen anhand von Fallstudien untersucht und vertieft. Lernmaterialien, Literaturangaben • Vorlesungsfolien und Übungsblätter • Aufgabenblätter und technische Dokumentation für die Rechnerübungen • Hennessey, Patterson: Computer Architecture: A Quantitative Approach (6th edition), Morgan Kaufmann, 2017 • Aktuelle Hinweise auf alternative und ergänzende Literatur, sowie Lehrmaterialien auf der Webseite und in den Vorlesungsfolien
----	--

3 Wahlpflichtmodule

Advanced Distributed Algorithms and Data Structures							
Advanced Distributed Algorithms and Data Structures							
Modulnummer: M.079.4006		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05700 Advanced Distributed Algorithms and Data Structures	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Advanced Distributed Algorithms and Data Structures:</i> Empfohlene Vorkenntnisse Algorithmen und Datenstrukturen, verteilte Algorithmen und Datenstrukturen						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Advanced Distributed Algorithms and Data Structures:</i> Nach einer kurzen Einführung in die Grundlagen der Graphen- und Netzwerktheorie sowie der verteilten Programmierung stellt die Vorlesung fortgeschrittene Methoden für verteilte Algorithmen und Datenstrukturen vor. Themen sind unter anderem Zugriffskontrolle, Synchronisation, Konsensus, Informationsverbreitung, hybride Netze, Scheduling und Optimierung. Aufbauend auf Lösungen zu diesen Themen werden auch konkrete Anwendungen vorgestellt. Die Vorlesung gibt eine Einführung in aktuelle, fortgeschrittene verteilte Algorithmen und Datenstrukturen. Neben der Präsentation der dafür notwendigen Protokolle werden diese auch rigoros auf ihre Korrektheit und Effizienz hin analysiert. Die Vorlesung gliedert sich in folgende Kapitel: • Einführung • Grundlagen der Graphen- und Netzwerktheorie • Zugriffskontrolle • Synchronisation • Konsensus • Informationsverbreitung • Hybride Netze • Scheduling • Optimierung Aufbauend auf Lösungen zu diesen Themen werden auch konkrete Anwendungen vorgestellt.						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können • grundlegende analytische Techniken verstehen und anwenden, • grundlegende algorithmische Ansätze erklären und anwenden, • beurteilen, welche Auswirkungen diese Ansätze haben und • die Grenzen der Anwendbarkeit dieser Ansätze einschätzen.			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch			
12	Modulbeauftragte/r: Prof. Dr. Christian Scheideler			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Advanced Distributed Algorithms and Data Structures:</i> Methodische Umsetzung Die Vorlesung nutzt Tafelanschrieb und Folien sowie kleine Aufgaben für die Studierenden während der Vorlesung. Sie wird durch Tutorien begleitet. Studierende haben in den Tutorien Gelegenheit, Aufgaben in der Gruppe zu bearbeiten und Lösungen von Übungsblättern mit den Tutoren zu diskutieren. Lernmaterialien, Literatur • Foliensatz der Vorlesung; Übungsblätter • Weitere Literatur wird in der Vorlesung bekanntgegeben
----	--

3 Wahlpflichtmodule

Advanced Networked Systems							
Advanced Networked Systems							
Modulnummer: M.079.4096		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
	Lehrveranstaltung		Lehr- form	Kontakt- zeit (h)	Selbst- studium (h)	Status (P/WP)	Gruppen- größe (TN)
	a)	L.079.05820 Advanced Networked Sys- tems	V2 Ü3	75	105	P	50/25
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Advanced Networked Systems:</i> Empfohlene Vorkenntnisse Kenntnisse in den Bereichen Computernetzwerke, Betriebssysteme, Programmiersprachen, C/C++ und Python-Programmierung in der Linux-Umgebung sowie ein ausgeprägtes Interesse daran, zu verstehen, wie die Dinge unter der Haube funktionieren. Fähigkeit, wissenschaftliche Arbeiten professionell zu lesen. Fähigkeit, in einem komplexen Umfeld zu programmieren.						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Advanced Networked Systems:</i> Der Kurs behandelt Konzepte und Entwürfe für moderne vernetzte Systeme, die vom Internet und von Cloud-Rechenzentren übernommen werden, um die ständig steigenden Anforderungen an die Datenübertragung und die von Big-Data- und Machine-Learning-Anwendungen angetriebenen Berechnungen zu erfüllen. • Grundlagen der Netzwerktechnik (Auffrischung) • Rechenzentrumsnetzwerke (Architekturen, Überlastungskontrolle) • Softwaredefinierte Netzwerke (SDN, OpenFlow) • Programmierbare Netzwerke (P4, eBPF/XDP) • Architekturen programmierbarer Netzwerkgeräte (RMT, SmartNICs) • Netzwerküberwachung und In-Network-Computing (Caching, Aggregation) • Systeme für maschinelles Lernen (Modelltraining, Systeme für LLMs)						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Nach Abschluss dieses Kurses werden die Studierenden in der Lage sein, • Kenntnisse über aktuelle Forschungsthemen im Bereich der vernetzten Systeme zu erlangen. • das Design dieser neuen Technologien für vernetzte Systeme zu verstehen und über die darin enthaltenen Designentscheidungen nachzudenken. • komplexe vernetzte Systeme unter Anwendung einiger dieser Entwürfe zu bauen, die Vorzüge und Grenzen dieser Entwürfe zu analysieren und zu bewerten und die Entwurfsentscheidungen für die gebauten Systeme zu erklären.											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><td>zu</td><td>Prüfungsform</td><td>Dauer bzw. Umfang</td><td>Gewichtung für die Modulnote</td></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: <table><tr><td>zu</td><td>Form</td><td>Dauer bzw. Umfang</td><td>SL / QT</td></tr><tr><td>a)</td><td>Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte</td><td></td><td>SL</td></tr></table>				zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
zu	Form	Dauer bzw. Umfang	SL / QT									
a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL									
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch											
12	Modulbeauftragte/r: Prof. Dr. Lin Wang											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Advanced Networked Systems:</i> Methodische Umsetzung Der Kursinhalt wird mit Hilfe von Folienvorlesungen, interaktiven Übungen und programmierbasierten Projektaufgaben vermittelt. Ein Großteil des Kurses wird auf Diskussionen über aktuelle Forschungsthemen basieren, die durch praktische Programmieraufgaben ergänzt werden. Lernmaterialien, Literaturangaben • Vorlesungsfolien und Übungsblätter • Projektbeschreibungsunterlagen für die Programmieraufgaben • Zusätzliche Literatur (z. B. Forschungsarbeiten) auf der Website der Lehrveranstaltung und in den Vorlesungsfolien
----	--

3 Wahlpflichtmodule

Agentic Software Engineering								
Agentic Software Engineering								
Modulnummer: M.079.4212		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester		
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en		
1	Modulstruktur							
		Lehrveranstaltung		Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05705 Agentic Software Engineering		V2 Ü3	75	105	P	80
2	Wahlmöglichkeiten innerhalb des Moduls: keine							
3	Teilnahmevoraussetzungen: keine							

4	Inhalte: <i>Inhalte der Lehrveranstaltung Agentic Software Engineering:</i> Seit dem Aufkommen generativer KI (GenAI), und insbesondere seit der Nutzung von GenAI im Rahmen von Agenten, befindet sich das Software Engineering (SE) in einem radikalen Wandel, der viele Fragen aufwirft: In welchen Phasen des SE lässt sich GenAI effektiv einsetzen? Was sind dabei die Chancen, aber auch die Limitierungen und Fallstricke? Wie verändert GenAI das Kompetenzprofil, das zukünftig von Software Engineers erwartet wird? In dieser Lehrveranstaltung wird der Einsatz von GenAI und Agenten in den verschiedenen Phasen des Softwarelebenszyklus (Anforderungsanalyse, Implementierung, Testing usw.) diskutiert, praktisch erlebbar gemacht und – wo angemessen – auch kritisch hinterfragt. Darüber hinaus wird aktuelle Forschung zu KI-gestützten Methoden der Softwareentwicklung behandelt. In praktischen Übungen werden die Studierenden an aktuelle Technologien herangeführt, die den Entwurf und den Einsatz von Agenten im Software Engineering ermöglichen oder vereinfachen. Behandelte Themen: • Grundlagen generativer KI (kompakter Recap als Ausgangspunkt) • Retrieval-Augmented Generation und Agenten: Funktionsweise und Abgrenzung beider Ansätze • Ökonomische Perspektive: Produktivitätseffekte (empirische Studien), Kosten, Latenz und Modell-Ökonomie beim produktiven Agenteneinsatz • Wissens- und Kontext-Engineering für SE-Aufgaben: Bereitstellung von Repositories, Issues, Tests und Dokumentation als Kontext (Informationskuratierung, Context Windows, Repo-Maps, RAG über Codebasen) • Tool-Use und Function Calling: Schnittstellen zwischen Agenten und Entwicklungswerkzeugen (Compiler, Test-Runner, Linter, Debugger, Versionsverwaltung) sowie Standards wie das Model Context Protocol (MCP) und das Agent Client Protocol (ACP) • Gängige Typen von Agenten (ReAct, Plan-and-Execute, Reflexion usw.) • Multi-Agenten-Systeme (inkl. Agent2Agent-Protokoll) • Nutzen und Herausforderungen von Agenten in den verschiedenen Phasen des Softwarelebenszyklus (Anforderungsanalyse, Implementierung, Testing usw.) • Feedback durch Testing und symbolische Verfahren (z. B. statische Analyse und Verifikation) als Korrektiv und Sicherheitsnetz • Verlässlichkeit und Grenzen: Halluzinationen, das Out-of-Distribution-Problem bei proprietären Codebasen, Korrektheitsgarantien • Sicherheit agentischer Systeme: Prompt Injection, autonome Ausführung, Supply-Chain-Risiken durch generierten Code • Human-in-the-loop, Verantwortung und Code-Ownership: veränderte Rollen- und Kompetenzprofile von Software Engineers
---	--

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Nach Abschluss des Moduls können die Studierenden • erklären, wie generative KI, Retrieval-Augmented Generation und agentische Systeme funktionieren, und ihre jeweiligen Einsatzbereiche im Software Engineering unterscheiden, • aktuelle Agenten-Technologien auf Aufgaben über den gesamten Softwarelebenszyklus anwenden und sie über Repositories, Tests und Werkzeugschnittstellen mit geeignetem Kontext versorgen, • den Nutzen und die Herausforderungen des Einsatzes von GenAI und Agenten in den einzelnen Phasen der Softwareentwicklung analysieren, • die Verlässlichkeit, Sicherheit und Grenzen agentischer Systeme beurteilen, einschließlich Halluzinationen, des Out-of-Distribution-Problems und von Korrektheitsgarantien, • eine eigene agentenbasierte Lösung für eine Aufgabe des Software Engineering entwerfen und die damit verbundenen Architekturentscheidungen begründen, • die ökonomischen und organisatorischen Auswirkungen eines produktiven Agenteneinsatzes bewerten, einschließlich der Effekte auf Kosten, Produktivität und die Rolle von Software Engineers.											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: <table><tr><th>zu</th><th>Form</th><th>Dauer bzw. Umfang</th><th>SL / QT</th></tr><tr><td>a)</td><td>Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte</td><td></td><td>SL</td></tr></table>				zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
zu	Form	Dauer bzw. Umfang	SL / QT									
a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL									
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Informatik v4											
12	Modulbeauftragte/r: Prof. Dr. Eric Bodden											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Agentic Software Engineering:</i> Methodische Umsetzung Die Inhalte werden im Rahmen einer Vorlesung präsentiert und in begleitenden Übungen vertieft. In praktischen Übungen wenden die Studierenden aktuelle Agenten-Technologien selbst an und erproben den Entwurf und Einsatz von Agenten an konkreten Aufgaben des Software Engineering. Aktuelle Forschungsergebnisse werden anhand von Originalliteratur diskutiert und kritisch eingeordnet. Lernmaterialien, Literaturangaben • Vorlesungsfolien und Übungsaufgaben • Grundlagenliteratur zu den behandelten Agenten-Architekturen • Aktuelle Forschungsliteratur und Konferenzbeiträge zu KI-gestütztem Software Engineering, die in der Veranstaltung bekanntgegeben werden • Spezifikationen und Dokumentation einschlägiger Protokolle (Model Context Protocol, Agent Client Protocol) • Weiterführende Literatur und Online-Ressourcen werden in der Veranstaltung bereitgestellt.
----	--

3 Wahlpflichtmodule

Algorithms for Complex Virtual Scenes							
Algorithms for Complex Virtual Scenes							
Modulnummer: M.079.4009		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05803 Algorithms for Complex Virtual Scenes	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Algorithms for Complex Virtual Scenes:</i> Empfohlene Vorkenntnisse Bereitschaft und Fähigkeit, den kreativen Prozess des Algorithmenentwurfs und die Effizienzanalyse mit mathematischen Methoden zu erlernen. Grundkenntnisse einiger grundlegender Algorithmen und Datenstrukturen und deren Analysen werden angenommen.						

3 Wahlpflichtmodule

4	Inhalte: <i>Inhalte der Lehrveranstaltung Algorithms for Complex Virtual Scenes:</i> Walkthrough-Systeme erlauben das Betrachten und Durchlaufen von virtuellen 3D-Szenen und finden Anwendung in Architekturprogrammen, Simulationen, oder Spielen. Die Effizienz von Echtzeit-Rendering Algorithmen ist entscheidend für eine flüssige und schnelle Darstellung der virtuellen 3D-Szenen in einem Walkthrough-System. Es gibt verschiedene algorithmische Ansätze, um hochkomplexe geometrische 3D-Daten zu reduzieren und eine Darstellung der Daten in Echtzeit zu erreichen. In der Vorlesung werden dafür unterschiedliche algorithmische Ansätze vorgestellt, z. B. Visibility-Culling, Simplification, Level of Detail, Image-Based Rendering. Die Veranstaltung umfasst folgende Inhalte: • Walkthrough-Problem • Räumliche Datenstrukturen: kd-Baum, BSP-Baum, Octree, Loose-Octree • Level of Detail: Adaptives LOD-Management, Mesh Simplification, Progressive Meshes • Visibility Culling: View Frustum Culling, Potentially Visible Sets (PVS), Dynamische Berechnung der PVS, Hierarchischer Z-Buffer, Coherent Hierarchical Culling, Hierarchische Occlusion Maps, Aspect-Graph, Visibility Space Partition • Replacement: Color-Cubes, Randomisierter Z-Buffer, Hierarchical Image Caching • Paralleles Rendern: Klassifizierung und Modellierung, Paralleles Rendering als Sortierproblem, Hybrides Sort-First/Sort-Last-Rendering								
5	Lernergebnisse und Kompetenzen: Die Studierenden können • grundlegende Algorithmen und Datenstrukturen für Problemstellungen der Darstellung komplexer virtueller Szenen benennen, erklären und anwenden • grundlegende algorithmische Problemstellungen in Anwendungsproblemen der Darstellung komplexer virtueller Szenen erkennen und geeignete Algorithmen und Datenstrukturen dafür auswählen • Laufzeit- und Speicherplatzabschätzung von räumlichen Datenstrukturen und Algorithmen analysieren, vergleichen und untersuchen • beurteilen, welche Auswirkungen die Wahl von räumlichen Datenstrukturen auf die Effizienz von Algorithmen der Darstellung komplexer virtueller Szenen hat • eigene effiziente Sichtbarkeits-Algorithmen auf Basis räumlicher Datenstrukturen für weitere virtuelle Szenen mit spezieller Charakteristik entwickeln • eigene effiziente Approximations-Algorithmen auf Basis räumlicher Datenstrukturen für weitere virtuelle Szenen mit spezieller Charakteristik entwickeln • Problemstellungen der Darstellung komplexer virtueller Szenen und entsprechende Lösungsvorschläge mit Fachexperten diskutieren								
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Mündliche Prüfung, Klausur oder Referat</td><td>120-180 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Mündliche Prüfung, Klausur oder Referat	120-180 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote						
a)	Mündliche Prüfung, Klausur oder Referat	120-180 min, 30-45 min oder 30 min	100%						

3 Wahlpflichtmodule

7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Informatik v4		
12	Modulbeauftragte/r: Dr. Matthias Fischer		

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Algorithms for Complex Virtual Scenes:</i> Methodische Umsetzung Die Inhalte werden durch eine Präsentation im Rahmen einer Vorlesung vermittelt. Die Vorlesung erfolgt in der Regel mit Beamer und Tafelanschrieb. In Übungen und Übungsaufgaben werden Entwurf und Analyse von Algorithmen an ausgewählten Beispielen vorgestellt und von den Studierenden in Präsenzübungen in Kleingruppen sowie im Selbststudium vertieft und selbst entwickelt und durch praktische Übungen ergänzt. Musterlösungen von Übungsblättern werden in Zentralübungen vorgestellt. Die erwarteten Aktivitäten der Studierenden sind die Mitarbeit bei Präsenzübungen und das selbstständige Bearbeiten der Übungsaufgaben. Lernmaterialien • Vorlesungsfolien, Übungsblätter, ggfs. Musterlösungen, Vorlesungsaufzeichnung aus früheren Jahren, Tafelabschrieb Literaturangaben • Real-Time Rendering; Tomas Akenine-Möller, Eric Haines; AK Peters, 2002. • Level of Detail for 3D Graphics; David Luebke, Martin Reddy, Jonathan D. Cohen; Morgan Kaufmann Publishers, 2002. • Algorithmen in der Computergraphik; Thomas Rauber; Teubner, 1993. • Wavelets for Computer Graphics: Theory and Applications; Eric Stollnitz, David H. Salesin, Anthony D. DeRose; Morgan Kaufmann Publishers, 1996. • Graphic Gems; Andrew S. Glassner; Academic Press; 1990. • Game Programming Gems; Mark DeLoura; Charles River Media; 2000. Algorithmische Geometrie • Computational Geometry - Algorithms and Applications; Mark de Berg, Marc de Kreveld, Mark Overmars; Springer Verlag, 2000. • Computational Geometry in C; Joseph O'Rourke; Cambridge University Press, 1998. • Algorithmic Geometry; Jean-Daniel Boissonnat, Herve Bronniman; Cambridge University Press, 1998. • Algorithmische Geometrie Grundlagen, Methoden, Anwendungen; Rolf Klein; Springer Verlag, 2005. Allgemeine Prinzipien der Computergrafik • 3D Computer Graphics; Alan Watt; Addison Wesley, 1999. • Computer Graphics, Principles and Practice; James Foley, Andries van Dam, Steven Feiner, John Hughes; Addison Wesley, 1995. • Computer Graphics; Donald Hearn, M. P. Baker; Prentice Hall, 2003. • Weitere Literatur wird in der Veranstaltung bekanntgegeben.
----	---

3 Wahlpflichtmodule

Approximate Computing							
Approximate Computing							
Modulnummer: M.079.4068		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.09758 Approximate Computing	V3 Ü2	75	105	P	100/30
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Approximate Computing:</i> Empfohlene Vorkenntnisse Kenntnisse aus den Bachelor-Modulen Digitaltechnik und Rechnerarchitektur sind hilfreich.						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Approximate Computing:</i> Approximate Computing ist ein aktueller Ansatz, der die Genauigkeit von Berechnungen reduziert und dadurch signifikante Einsparungen im Energieverbrauch, der Rechenzeit oder der Chipfläche erzielt. Dieser forschungsorientierte Kurs führt in das Gebiet des Approximate Computing ein und stellt die wesentlichen Methoden für die Implementierung effizienter Rechnersysteme durch Reduktion der Genauigkeit vor. Der Kurs behandelt Approximationstechniken auf allen Ebenen eines Rechnersystems, von der Anwendungsebene bis hin zur Ebene der Hardwaretechnologie. In den Übungen/Tutorium wird die Effizienz dieser Techniken für verschiedene Anwendungsdomänen, wie zum Beispiel Deep Learning und Digital Signal Processing, untersucht. • Einführung und Motivation für ungenaues Rechnen • Approximation in der Anwendungsebene, z.B. für maschinelles Lernen und digitale Signalverarbeitung • Programmiersprachen/Compiler für Approximate Computing • Approximation in der Mikroarchitektur • Automatisierte Synthese von approximierten Schaltungen • Ungenaue arithmetische Komponenten und Performanceoptimierung durch Reduktion der Genauigkeit • Approximationstechniken in der Technologieebene • Übungen/Tutorial: Approximation von Algorithmen des Deep Learning und Digital Signal Processing						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden sind nach dem Besuch der Lehrveranstaltung in der Lage, • die Approximationstechniken auf den unterschiedlichen Ebenen eines Rechnersystems zu benennen und zu erklären, • die wesentlichen technischen/wissenschaftlichen Problemstellungen bei der Approximation von Rechnersystemen zu identifizieren, • die Einsetzbarkeit der Approximationstechniken für verschiedene Anwendungsdomänen zu beurteilen und • die Approximationstechniken anzuwenden, um effiziente Hardwarebeschleuniger zu realisieren, insbesondere für Deep Learning und Digital Signal Processing											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: <table><tr><th>zu</th><th>Form</th><th>Dauer bzw. Umfang</th><th>SL / QT</th></tr><tr><td>a)</td><td>Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte</td><td></td><td>SL</td></tr></table>				zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
zu	Form	Dauer bzw. Umfang	SL / QT									
a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL									
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Informatik v3											
12	Modulbeauftragte/r: Prof. Dr. Marco Platzner											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Approximate Computing:</i> Methodische Umsetzung • Vorlesung mit Beamer und Tafelanschrieb • Interaktive Übungen/Diskussionen im Hörsaal • Rechnerübungen Lernmaterialien, Literaturangaben • Vorlesungsfolie und Aufgabenblätter • Adrian Sampson, Luis Ceze, and Dan Grossman: Good-Enough Computing. IEEE Spectrum, 50(10):54-59, 2013 • Ravi Nair. Big Data Needs Approximate Computing: Technical Perspective. Communications of the ACM, 58(1): 104, 2015. • Sparsh Mittal. A Survey of Techniques for Approximate Computing. ACM Computing Surveys, 48(4), 2016. • Qiang Xu, Todd Mytkowitz, and Nam Sung Kim. Approximate Computing: A Survey. IEEE Design & Test, 33(1):8-22, 2016. • Weiqiang Liu and Fabrizio Lombardi (Editors), Approximate Computing. Springer, 2022. • Zusätzliche Ressourcen und Links auf aktuelle Publikationen werden in der Vorlesung zur Verfügung gestellt.
----	---

3 Wahlpflichtmodule

Data-Driven Innovation								
Data-Driven Innovation								
Modulnummer: M.079.4076		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester		
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en		
1	Modulstruktur							
		Lehrveranstaltung		Lehr- form	Kontakt- zeit (h)	Selbst- studium (h)	Status (P/WP)	Gruppen- größe (TN)
	a)	L.079.05822 Data-Driven Innovation		V2 Ü3	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine							
3	Teilnahmevoraussetzungen: keine							

4	Inhalte: <i>Inhalte der Lehrveranstaltung Data-Driven Innovation:</i> Innovationen entstehen, wenn Unternehmen erfolgreich neue oder verbesserte Produkte und Dienstleistungen auf den Markt bringen. Innovationen sind die Voraussetzung für Wohlstand, Wirtschaftswachstum und Wettbewerbsvorteile. Die zunehmende Digitalisierung aller Lebensbereiche hat zu zahlreichen neuen Ansätzen für Innovationen und deren Entwicklung geführt. Die Vorlesung Data-Driven Innovation gibt dazu einen grundlegenden Überblick: Es werden die Grundlagen des Innovations- und Datenmanagements besprochen, neue digitalisierte Marktleistungen und Geschäftsmodelle erörtert und die Auswirkungen auf den Innovationsprozess untersucht und analysiert. Darauf aufbauend beschäftigt sich die Vorlesung mit der Frage, wie Organisationen datengetriebene Innovationen umsetzen können. Das Modul umfasst die folgenden Inhalte: • Innovation und Daten – Grundlagen des Innovationsmanagements – Grundlagen von Daten in Organisationen • Datengesteuerte Angebote – Intelligente Produkte – Intelligente Dienstleistungen – Digitale Plattformen – Data Spaces & Digitale Geschäftsmodelle • Innovationsprozesse – Methoden und Werkzeuge des Innovationsmanagements – Datengetriebene Innovationsprozesse • Organisation der datengetriebenen Transformation – Digitale Transformation – Daten für Nachhaltigkeit nutzen
5	Lernergebnisse und Kompetenzen: Die Studierenden • die Grundlagen des Innovations- und Datenmanagements sowie wesentliche Konzepte und Ansätze kennenlernen • verstehen, welchen Einfluss die Digitalisierung auf die Marktleistungen insbes. produzierender Unternehmen hat • in der Lage sein, Innovationsprozesse in der Praxis zu verstehen und zu reflektieren. • nachvollziehen, wie die Transformation zum datengetriebenen Unternehmen erfolgen kann • diverse Ansätze zur Analyse von Problemen und zur Lösungsfindung anwenden können. • in der Lage sein, Ideen systematisch zu finden, zu konzipieren, zu erproben und in Richtung Marktangang zu entwickeln.

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme: keine		
8	Voraussetzungen für die Teilnahme an Prüfungen: keine		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Master's Program Electrical Systems Engineering (ESEMA v2), Master's Program Electrical Systems Engineering v3 (ESEMA v3)		
12	Modulbeauftragte/r: Dr. Christian Koldewey, Prof. Dr.-Ing. Roman Dumitrescu		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Data-Driven Innovation:</i> Methodische Umsetzung Das Modul besteht aus zwei Teilen: 1. Vorlesung mit Folien: Grundlagen und Konzepte werden in der Vorlesung erläutert und mit Beispielen illustriert. 2. Übungen (Tutorium): In den Übungen wird das Wissen vermittelt und die Konzepte werden angewandt. Die Übungen müssen von den Studierenden selbst vorbereitet werden. Lernmaterialien, Literaturangaben • Gausemeier, Jürgen & Dumitrescu, Roman & Echterfeld, Julian & Pfänder, Thomas & Steffen, Daniel & Thielemann, Frank. (2018). Innovationen für die Märkte von morgen: Strategische Planung von Produkten, Dienstleistungen und Geschäftsmodellen. 10.3139/9783446429727. (https://www.hanser-fachbuch.de/fachbuch/artikel/9783446428249#content-desc) • Beverungen, Daniel & Dumitrescu, Roman & Kühn, Arno & Plass, Christoph. (2024). Digitale Plattformen im industriellen Mittelstand Strategien, Methoden, Umsetzungsbeispiele. • https://ki-marktplatz.com/wp-content/uploads/2021/02/KI-MP_Whitepaper.pdf • https://www.advanced-systems-engineering.de/#studie		

3 Wahlpflichtmodule

Data Science for Software Engineering							
Data Science for Software Engineering							
Modulnummer: M.079.4101		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05817 Data Science for Software Engineering	V2 Ü3	75	105	P	30
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: <i>Teilnahmevoraussetzungen der Lehrveranstaltung Data Science for Software Engineering:</i> Empfohlene Vorkenntnisse Gute Programmierkenntnisse in Java und/oder Python sind hilfreich, um die Aufgaben zu erledigen. Grundlegende Hintergrundinformationen zum maschinellen Lernen sind hilfreich, um einige der Data-Science-Konzepte zu verstehen.						

3 Wahlpflichtmodule

4	Inhalte: <i>Inhalte der Lehrveranstaltung Data Science for Software Engineering:</i> Softwareentwickler befassen sich bei ihrer täglichen Arbeit mit Software-Repositories, etwa wenn sie Quellcode in Versionsverwaltungssystemen schreiben, Issues in Issue-Trackern posten, per E-Mail in Mailinglisten kommunizieren oder in Foren und Blogs diskutieren. Die großen Datenmengen in Software-Repositories, ihre kontinuierliche Weiterentwicklung, Komplexität und Heterogenität stellen eine Herausforderung für Softwareentwickler dar. In den vergangenen Jahren haben Forscher Ansätze vorgeschlagen, die Methoden aus der Data Science Wissenschaftsfeld verwenden, um Softwareentwickler zu unterstützen. In dieser Lehrveranstaltung wird die Anwendung von Data-Science-Methoden auf Software- Repositorien erläutert, um allgemeine Software-Engineering-Aufgaben zu lösen. Die Lehrveranstaltung umfasst folgende Themen: • Arten und Struktur von Software-Repositories. • Clustering von Quellcode. • Pipeline zur Verarbeitung natürlicher Sprache. • Themenmodellierung • Worteinbettung. • Information retrieval. • Überwachtes maschinelles Lernen. • Statistische Analyse. Konzepte werden in den Vorlesungen besprochen und mithilfe einer Reihe von Gruppenaufgaben angewendet, um Open-Source-Systeme zu analysieren und bestimmte Softwarearchitektur- und Wartungsaufgaben zu erfüllen.								
5	Lernergebnisse und Kompetenzen: Die Studierenden können • Arten und Struktur von Software-Repositories klären und diskutieren. • die Hauptkonzepte von Data Science Methoden und deren Anwendung auf Software-Repositories klären und diskutieren. • Data Science Methoden auf große Software-Repositories anwenden. • aus den Analyseergebnissen nützliche Implikationen ableiten. • analyseergebnisse in einem wissenschaftlichen Format zusammenfassen und berichten. • in Gruppen arbeiten. • einen fachlichen Vortrag halten. • wissenschaftlich schreiben.								
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur oder mündliche Prüfung</td><td>90-120 min bzw. 30-45 min</td><td>100%</td></tr></table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur oder mündliche Prüfung	90-120 min bzw. 30-45 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote						
a)	Klausur oder mündliche Prüfung	90-120 min bzw. 30-45 min	100%						

3 Wahlpflichtmodule

7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben und Kurzreferate	SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Informatik v3		
12	Modulbeauftragte/r: Dr. Mohamed Aboubakr Mohamed Soliman		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Data Science for Software Engineering:</i> Methodische Umsetzung Der Schwerpunkt des Kurses liegt mehr auf der Anwendung von Data-Science-Methoden in der Softwareentwicklung als auf dem mathematischen Hintergrund von Data-Science-Methoden. Die Konzepte der Methoden werden durch eine Präsentation im Rahmen einer Vorlesung vermittelt und die Anwendung der Methoden durch Gruppenaufgaben und Präsentationen vertieft. Lernmaterialien, Literaturangaben Neben den Folien werden zu jedem Thema weitere Lernmaterialien aus prominenten Publikationen der Software-Engineering-Literatur bereitgestellt.		

3 Wahlpflichtmodule

Designing code analyses for large-scale software systems 1							
Designing code analyses for large-scale software systems 1							
Modulnummer: M.079.4070		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05781 Designing code analyses for large-scale software systems 1	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Designing code analyses for large-scale software systems 1:</i> Empfohlene Vorkenntnisse Ein gutes Verständnis von Java und den Prinzipien objektorientierter Programmierung ist hilfreich.						

4	Inhalte: <i>Inhalte der Lehrveranstaltung Designing code analyses for large-scale software systems 1:</i> Statische Codeanalysen dienen u.a. dazu, automatisiert Fehler und Schwachstellen im Programmcode aufzufinden. Zu diesem Zwecke suchen sie nach bekannten Fehlermustern. In dieser Vorlesung wird erklärt, wie man solche Codeanalysen entwirft, die inter-prozedural sind, also das komplette Programm betrachten, über die Grenzen einzelner Prozeduren hinweg. Der Entwurf solcher Analysen gestaltet sich deshalb sehr schwierig, weil die Analysen oft Millionen von Programmstatements gleichermaßen präzise aber auch effizient verarbeiten müssen. Es werden außerdem Beispielsanalysen aus dem Bereich der IT-Sicherheit besprochen. Diese Veranstaltung ist Teil einer Kombination DECA 1/2. In DECA 2 werden aktuelle Ansätze aus der Forschung besprochen. Es wird dringend empfohlen zuerst DECA 1 und dann DECA 2 zu belegen. Behandelte Themen: • Typsysteme und fluss-insensitive Analysen • Endliche Verbände und Fixpunkte • Intra-prozedurale fluss-sensitive Codeanalysen • Intervallanalyse, Widening und Narrowing • Erstellen von Call-graphen • Pointer-Analyse • Inter-prozedurale Codeanalysen • Context-sensitive Analyse mit dem Call-strings Approach • Context-sensitive Analyse mit dem Functional approach • Value-based Termination, VASCO • Distributive Analysen mit IFDS • Praktische Definitionen von Flussfunktionen • Distributive Analysen mit IDE Während der gesamten Veranstaltung werden Anwendungsbeispiele aus dem Gebiet der Softwaresicherheit diskutiert.
5	Lernergebnisse und Kompetenzen: Nach Abschluss des Moduls können die Studierenden • die wichtigsten Konzepte und Algorithmen im Bereich der statischen Programmanalyse benennen und unterscheiden • erklären, welche Auswirkungen verschiedene alternativer Entwurfsentscheidungen beim Entwurf einer statischen Programmanalyse nach sich ziehen • einfachere statische Programmanalysen selbst implementieren und anwenden sowie deren Funktion illustrieren • Datenstrukturen und Algorithmen zur statischen Programmanalyse gegenüberstellen und vergleichen • die Anwendbarkeit bestimmter Analyseverfahren auf bestimmte Anwendungskontexte bewerten und begründen sowie • durch Komposition mehrerer Analyseverfahren Werkzeuge zur statischen Programmanalyse selbst entwickeln.

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL / QT
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch		
12	Modulbeauftragte/r: Prof. Dr. Eric Bodden		

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Designing code analyses for large-scale software systems 1:</i> Methodische Umsetzung Vorlesung und Gruppenübungen sowie praktische Programmierübungen mit weltweit genutzten Frameworks für die statische Codeanalyse Lernmaterialien, Literaturangaben • Thomas Reps, Susan Horwitz, and Mooly Sagiv. 1995. Precise interprocedural dataflow analysis via graph reachability. POPL '95 • Shmuel Sagiv, Thomas W. Reps, and Susan Horwitz. 1995. Precise Interprocedural Dataflow Analysis with Applications to Constant Propagation. TAPSOFT '95 • Akash Lal, Thomas Reps, and Gogul Balakrishnan. 2005. Extended weighted pushdown systems. CAV 2005 • Nomair A. Naeem, Ondrej Lhoták, and Jonathan Rodriguez. 2010. Practical extensions to the IFDS algorithm. CC 2010 • Yannis Smaragdakis, Martin Bravenboer, and Ondrej Lhoták. 2011. Pick your contexts well: understanding object-sensitivity. POPL 2011 • Eric Bodden. 2012. Inter-procedural data-flow analysis with IFDS/IDE and Soot. SOAP 2012 • Rohan Padhye, Uday P. Khedker. Interprocedural Data Flow Analysis in Soot using Value Contexts. SOAP 2013
----	--

3 Wahlpflichtmodule

Designing code analyses for large-scale software systems 2							
Designing code analyses for large-scale software systems 2							
Modulnummer: M.079.4071		Workload (h): 180	Leistungspunkte: 6		Turnus: Sommersemester		
		Studiensemester: 2-3	Dauer (in Sem.): 1		Sprache: en		
1	Modulstruktur						
	Lehrveranstaltung		Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
a)	L.079.05800 Designing code analyses for large-scale software systems 2		V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine Teilnahmevoraussetzungen der Lehrveranstaltung Designing code analyses for large-scale software systems 2: Empfohlene Vorkenntnisse Der vorherige Besuch der Veranstaltung DECA 1 wird dringend empfohlen. Ein gutes Verständnis von Java und/oder C++ und den Prinzipien objektorientierter Programmierung ist hilfreich.						

4	Inhalte: <i>Inhalte der Lehrveranstaltung Designing code analyses for large-scale software systems 2:</i> Statische Codeanalysen dienen dazu, automatisiert Fehler und Schwachstellen im Programmcode aufzufinden. Zu diesem Zwecke suchen sie nach bekannten Fehlermustern. In dieser Vorlesung wird erklärt, wie man solche Codeanalysen entwirft, die inter-prozedural sind, also das komplette Programm betrachten, über die Grenzen einzelner Prozeduren hinweg. Der Entwurf solcher Analysen gestaltet sich deshalb sehr schwierig, weil die Analysen oft Millionen von Programmstatements gleichermaßen präzise aber auch effizient verarbeiten müssen. Es werden außerdem Beispielsanalysen aus dem Bereich der IT-Sicherheit besprochen. Diese Lehrveranstaltung knüpft an an die Veranstaltung DECA 1. In DECA 2 werden vor allem neuartige Konzepte direkt aus der Forschung besprochen, beispielsweise sogenannte demand-driven analyses, welche sich durch eine präzisere und gleichzeitig effizientere Analyse auszeichnen, aber auch Pushdown-Systeme, die eine elegante Modellierung und ebenso schnelle Ausführung von Programmanalysen erlauben. Zu guter letzt erklären wir aktuelle Lösungsansätze zu praktischen Problemen in der statischen Analyse wie beispielsweise der Nutzung von Reflection und nativem Code. Behandelte Themen: • Programmanalyse von Software-Produktlinien • Modellierung von Call Stacks und Feldzugriffen mit Pushdown-Systemen • Modellierung von weiterer Analyseinformationen mit Weighted Pushdown Systems • Effizienz- und Präzisionsgewinne durch bedarfsgesteuerte Programmanalyse • Synchronisierte Pushdown-Systeme im Boomerang-Framework • Angewandte Android-Code-Analyse mit FlowDroid • Behandlung von Reflexion mittels TamiFlex • Hybride statische und dynamische Analyse mit Harvester • Lernen von Quell-, Senken- und Sanitizer-Definitionen mit SWAN und SWAN Assist • Erklärbare statische Analyse
5	Lernergebnisse und Kompetenzen: Nach Abschluss des Moduls können die Studierenden • die wichtigsten Herausforderungen bei der inter-prozeduralen statischen Programmanalyse benennen und erklären • aktuelle Verfahren im Bereich der inter-prozeduralen statischen Programmanalyse benennen und unterscheiden • erklären, welche Auswirkungen verschiedene alternativer Entwurfsentscheidungen beim Entwurf einer inter-prozeduralen statischen Programmanalyse nach sich ziehen • Datenstrukturen und Algorithmen zur inter-prozeduralen statischen Programmanalyse gegenüberstellen und vergleichen • die Anwendbarkeit aktueller Analyseverfahren in einer breiten Auswahl von Anwendungskontexten bewerten und begründen.

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL / QT
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch		
12	Modulbeauftragte/r: Prof. Dr. Eric Bodden		

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Designing code analyses for large-scale software systems 2:</i> Methodische Umsetzung Vorlesung und Gruppenübungen sowie Programmierübungen mittels realer, weltweit genutzter Frameworks für die statische Analyse (bspw. Soot, Phasar, FlowDroid) Lernmaterialien, Literaturangaben • Context-, Flow-, and Field-sensitive Data-flow Analysis Using Synchronized Pushdown Systems (Johannes Späth, Karim Ali, Eric Bodden), In Proceedings of the ACM SIGPLAN Symposium on Principles of Programming Languages, pages 48:1–48:29, 3(POPL), 2019. • FlowDroid: Precise Context, Flow, Field, Object-sensitive and Lifecycle-aware Taint Analysis for Android Apps (Steven Arzt, Siegfried Rasthofer, Christian Fritz, Eric Bodden, Alexandre Bartel, Jacques Klein, Yves Le Traon, Damien Ochteau, Patrick McDaniel), In Proceedings of the 35th ACM SIGPLAN Conference on Programming Language Design and Implementation, pages 259–269, PLDI '14, ACM, 2014. • Codebase-Adaptive Detection of Security-Relevant Methods (Goran Piskachev, Lisa Nguyen Quang Do, Eric Bodden), In ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA), 2019. • Taming Reflection: Aiding Static Analysis in the Presence of Reflection and Custom Class Loaders (Eric Bodden, Andreas Sewe, Jan Sinschek, Hela Oueslati, Mira Mezini), In ICSE '11: International Conference on Software Engineering, pages 241–250, ACM, 2011.
----	---

3 Wahlpflichtmodule

Digitale Sprachsignalverarbeitung							
Digital Speech Signal Processing							
Modulnummer: M.048.24001		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1.-3. Semester		Dauer (in Sem.): 1		Sprache: de / en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.048.24001 Digitale Sprachsignalverarbeitung	2V 2Ü, SS	60	120	P	40/40
2	Wahlmöglichkeiten innerhalb des Moduls: Keine						
3	Teilnahmevoraussetzungen: Keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Digitale Sprachsignalverarbeitung:</i> Empfohlen: Vorkenntnisse aus dem Modul Höhere Mathematik.						

4	Inhalte: <i>Inhalte der Lehrveranstaltung Digitale Sprachsignalverarbeitung:</i> Kurzbeschreibung Die Veranstaltung führt in die grundlegenden Techniken und Theorien zur digitalen Sprachsignalverarbeitung ein. Schwerpunkt des ersten Teils der Vorlesung liegt im Themengebiet „Hören und Sprechen“, welches sich mit psychologischen Effekten der Geräuschwahrnehmung und der Spracherzeugung beschäftigt. Anschließend werden zeitdiskrete Signale und Systeme, sowie deren rechnergestützte Verarbeitung besprochen. Die nichtparametrische Kurzeitanalyse von Sprachsignalen, die Sprachcodierung und die IP-Telefonie sind weitere Themen. Inhalt • Sprechen und Hören • Spracherzeugung: menschliche Sprechorgane, Lautklassen, Quelle-Filter-Modell, Vocoder • Grundlagen Schallwellen • Hören: menschliches Hörorgan, Psychoakustik und Physiologie des Hörens, Lautheit, Verdeckung, Frequenzgruppen • Zeitdiskrete Signale und Systeme • Grundlagen: Elementare Signale, LTI-Systeme • Transformationen: Fouriertransformation zeitdiskreter Signale, DFT, FFT • Realisierung zeitdiskreter Filterung im Frequenzbereich: Overlap-Add, Overlap-Save • Statistische Sprachsignalanalyse • Grundlagen Wahrscheinlichkeitsrechnung • Kurzeitanalyse von Sprachsignalen: Spektrogramm, Cepstrum • Schätzung von Sprachsignalen • Optimale Filterung • LPC-Analyse • Spektrale Filterung zur Rauschunterdrückung • Adaptive Filterung: LMS Adaptionsalgorithmus, Echokompensation • Sprachcodierung • Signalformcodierung, parametrische Codierung, hybride Codierv Verfahren • Codierung im Frequenzbereich • Amplitudenquantisierung: gleichförmige Quantisierung, Quantisierung mit Kompondierung (ulaw, alaw)
5	Lernergebnisse und Kompetenzen: Fachkompetenz: Die Studierenden sind nach dem Besuch der Lehrveranstaltung in der Lage, • Digitale Signale, speziell Audiosignale, im Zeit- und Frequenzbereich zu analysieren, • Sprachsignale effizient zu repräsentieren und • Weit verbreitete Algorithmen zur Sprachsignalanalyse und Verarbeitung im Frequenz- oder Zeitbereich zu implementieren. Fachübergreifende Kompetenzen: Die Studierenden • können Effekte in echten Signalen durch theoretisches Wissen erklären, • können theoretische Ansätze durch systematische Betrachtung untersuchen und • sind durch die fundierte Betrachtung der Inhalte in der Lage, sich selbst weiterzubilden

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)										
	<table> <tr> <th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr> <tr> <td>a)</td><td>Klausur oder mündliche Prüfung oder Referat</td><td>120-180 min oder 30-45 min oder 30 min</td><td>100%</td></tr> </table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur oder mündliche Prüfung oder Referat	120-180 min oder 30-45 min oder 30 min	100%		
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote								
a)	Klausur oder mündliche Prüfung oder Referat	120-180 min oder 30-45 min oder 30 min	100%								
7	Studienleistung, qualifizierte Teilnahme: keine										
8	Voraussetzungen für die Teilnahme an Prüfungen: Keine										
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe der Leistungspunkte erfolgt, wenn die Modulabschlussprüfung (MAP) bestanden ist.										
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Credits gewichtet (Faktor 1).										
11	Verwendung des Moduls in anderen Studiengängen: BF Informationstechnik Lehramt BK affine Fächer Master v5, Masterstudiengang Computer Engineering v3 (CEMA v3), Masterstudiengang Computer Engineering v3 (CEMA v3), englisch, Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Elektrotechnik v4 (EMA v4), Masterstudiengang Elektrotechnik v5 (EMA v5), Masterstudiengang Informatik v3, Masterstudiengang Wirtschaftsingenieurwesen Studienrichtung Elektrotechnik, Masterstudiengang Wirtschaftsingenieurwesen Studienrichtung Elektrotechnik V4										
12	Modulbeauftragte/r: Dr.-Ing. Jörg Schmalenströer										
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Digitale Sprachsignalverarbeitung:</i> Lehrveranstaltungsseite https://ei.uni-paderborn.de/nt/lehre/veranstaltungen/digitale-sprachsignalverarbeitung Methodische Umsetzung • Vorlesungen mit Tafelinsatz und Präsentationen, • Abwechselnde theoretische und praktische Präsenzübungen mit Übungsblättern und Rechnern und • Demonstrationen von echten Systemen in der Vorlesung Lernmaterialien, Literaturangaben Bereitstellung eines Skripts; Hinweise auf Lehrbücher; Matlab Skripte										

3 Wahlpflichtmodule

Explainable Artificial Intelligence							
Explainable Artificial Intelligence							
Modulnummer: M.079.4091		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05807 Explainable Artificial Intelligence	V2 Ü1 P2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Explainable Artificial Intelligence:</i> Empfohlene Vorkenntnisse Grundkenntnisse in maschinellem Lernen und Programmierung						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Explainable Artificial Intelligence:</i> Die Vorhersagen von Modellen des maschinellen Lernens zu erklären wird für immer mehr Anwendungen wichtig. Bankkunden möchten zum Beispiel wissen, warum ihr Kredit abgelehnt wurde; Entwickler möchten ihre Modelle debuggen und verbessern; Manager möchten die Einhaltung von gesetzlichen Vorschriften sicherstellen. Dieser Kurs zielt darauf ab, die Vorhersagen von Modellen des maschinellen Lernens zu erklären, und stellt verschiedene Erklärungsmethoden vor. Erklärungsmethoden können danach unterschieden werden, ob sie spezifisch für ein bestimmtes Modell oder modellunabhängig sind und ob sie eine einzelne Vorhersage oder das gesamte Modell erklären. • Einführung (z.B. Wichtigkeit der Interpretierbarkeit, Evaluierung der Interpretierbarkeit, Datensätze für Fallstudien) • Interpretierbare Modelle (z. B. lineare Regression, logistische Regression, Entscheidungsbäume, Entscheidungsregeln) • Globale modell-agnostische Methoden (z. B., Partial Dependence Plots, Permutation Feature Importance, Global Surrogate Models) • Lokale modell-agnostische Methoden (z. B. LIME, SHAP, Anchors, kontrafaktische Erklärungen) • Modell-spezifische Methoden (z. B. für neuronale Netze)						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Nach Abschluss des Moduls sind die Studierenden in der Lage • die Wichtigkeit der Interpretierbarkeit zu erkennen und zu diskutieren • wichtige Erklärungsmethoden (z.B. interpretierbare Modelle, modell-agnostische Methoden und modell-spezifische Methoden) zu erklären und anzuwenden • Charakteristika von Datensätzen, Machine-Learning-Aufgaben und Machine-Learning-Modellen in Anwendungsproblemen zu erkennen und zu argumentieren, welche Erklärungsmethode für ein bestimmtes Problem geeignet ist • einfache Erklärungsmethoden von Grund auf zu implementieren • vorhandene Erklärungsmethoden zu erweitern und zu modifizieren • Probleme und Lösungsvorschläge mit Experten auf dem Gebiet zu diskutieren • Forschungsliteratur aus dem XAI Bereich zu lesen und zu diskutieren											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: <table><tr><th>zu</th><th>Form</th><th>Dauer bzw. Umfang</th><th>SL / QT</th></tr><tr><td>a)</td><td>Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte</td><td></td><td>SL</td></tr></table>				zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
zu	Form	Dauer bzw. Umfang	SL / QT									
a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL									
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch											
12	Modulbeauftragte/r: Dr. Stefan Heindorf											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Explainable Artificial Intelligence:</i> Methodische Umsetzung Folien und Tafelanschrieb. Wichtige Konzepte und Techniken werden durch Übungen im Vorlesungsraum und in den Tutorien geübt und in einem Miniprojekt angewendet. Lernmaterialien, Literaturangaben • Folien • Übungen • Buch: Christoph Molnar. Interpretable machine learning. 2020. • Zusätzliches Material und Literatur wird in der Lehrveranstaltung bekannt gegeben.
----	--

3 Wahlpflichtmodule

Foundational Methods for Knowledge Representation and Reasoning							
Foundational Methods for Knowledge Representation and Reasoning							
Modulnummer: M.079.4210		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05811 Parameterized Algorithms and Complexity	V3 Ü2	75	105	P	35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Parameterized Algorithms and Complexity:</i> Empfohlene Vorkenntnisse Grundlegende Kenntnisse zu Algorithmen und ihrer Analyse, Komplexitätstheorie (Komplexitätsklassen, Turing-Maschinen, NP-Vollständigkeit) sowie Prädikatenlogik						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Parameterized Algorithms and Complexity:</i> Dieser Kurs gibt eine Einführung in die Theorie der parametrisierten Komplexität. Er • gibt eine Einführung in die Grundlegenden Ideen der parametrisierten Komplexität, • behandelt verschiedene algorithmische Techniken, die dabei helfen, effiziente Algorithmen (in Hinblick auf ihre parametrisierte Komplexität) zu konstruieren, beispielsweise Kernelisierung und beschränkte Suchbäume, • behandelt zentrale Konzepte und Methoden der Theorie der Schwere von Problemen in der parametrisierten Komplexität, insbesondere fpt-Reduktionen sowie die W-Hierarchie und A-Hierarchie, • gibt eine kurze Einführung in die fein-granulare Komplexität.						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können • in der Vorlesung behandelte Begriffe kompetent verwenden und erläutern • in der Vorlesung behandelte Beweise und Beweistechniken wiedergeben und erläutern, inklusive algorithmischer Techniken • insbesondere können sie die Beweise für Korrektheit und Laufzeit der behandelten Algorithmen wiedergeben und erläutern • konstruktive Beweise und Verfahren aus der Vorlesung auf konkrete Instanzen anwenden • die Beweistechniken und algorithmischen Techniken aus der Vorlesung auf neue, ähnliche Probleme anwenden			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Referat		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Informatik v3			
12	Modulbeauftragte/r: Prof. Dr.-Ing. Anni-Yasmin Turhan			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Parameterized Algorithms and Complexity:</i> Methodische Umsetzung Die Inhalte werden via Beamer oder am Whiteboard präsentiert. In den Übungen gibt es die Möglichkeit, das Verständnis der Inhalte zu vertiefen und die Zielkompetenzen des Kurses zu üben. Die Vorlesung folgt größtenteils ausgewählten Kapiteln der folgenden Literatur: • <i>Parameterized Algorithms</i> by M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, S. Saurabh (Springer 2015). • <i>Parameterized Complexity Theory</i> by J. Flum, M. Grohe (Springer 2006). • Weitere Literatur wird ggf. in der Veranstaltung bekanntgegeben.
----	---

3 Wahlpflichtmodule

Foundations of Cryptography							
Foundations of Cryptography							
Modulnummer: M.079.4020		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05801 Foundations of Cryptography	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Foundations of Cryptography:</i> Empfohlene Vorkenntnisse Basiskonntnisse in IT-Sicherheit und Kryptographie nützlich aber nicht notwendig, Grundkonzepte der Komplexitätstheorie und Wahrscheinlichkeitstheorie						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Foundations of Cryptography:</i> Kryptographie ist eine wichtige Basistechnik der IT-Sicherheit. So beruhen Internet-Protokolle wie TLS auf kryptgraphischen Primitiven wie Schlüsselaustausch, Verschlüsselung und Signaturen. In dieser Vorlesung werde wichtige Basiskonzepte moderner Kryptographie vorgestellt. Hierzu gehören Verschlüsselungsverfahren, digitale Signaturen, Identifikationsprotokolle und Mehrparteienberechnungen. In allen Fällen werden formale Sicherheitsdefinitionen vorgestellt und, ausgehend von mathematisch präzisen Annahmen, beweisbar sichere Konstruktionen entwickelt. Eine wesentlicher Aspekt der Vorlesung ist die Konstruktiver effizienter und sicherer kryptographischer Verfahren aus möglichst allgemeinen Annahmen. Zu den Inhalten gehören: • Symmetrische und asymmetrische Verschlüsselung • Pseudozufallsfunktionen, Einweg-Funktionen, Permutationen mit Falltüren • Hashfunctions und Authentifizierungscodes • Digitale Unterschriften, Einmal-Unterschriften und Zufallsorakel • Identifikationsprotokolle, Σ-Protokolle • Sicherheitskonzepte wie unfälschbare Signaturen und CPA- sowie CCA-sichere Verschlüsselungsverfahren						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Nach Abschluss des Moduls sind die Studierenden in der Lage: • Konzepte und Methoden moderner Kryptographie zu verstehen, zu erklären und anzuwenden. • gemäß den Sicherheitsanforderungen einer Anwendung geeignete kryptographischen Verfahren auszuwählen, z.B. unterscheiden, wo Verschlüsselungsverfahren und wo Authentisierungsverfahren angemessen sind. • Primitiven der Kryptographie gemäß den Anforderungen von Anwendungen kombinieren und die Sicherheit der Kombination zu beweisen. • neue Sicherheitskonzepte zu definieren und kryptographische Methoden zu entwerfen, die diesen Konzepte erfüllen. • Sicherheitsbeweise verstehen und eigenständig ausarbeiten. • sich neueste Forschungsergebnisse im Bereich der Kryptographie anhand von wissenschaftlichen Papers zu erarbeiten.								
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>120-180 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote						
a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%						
7	Studienleistung, qualifizierte Teilnahme: <table><tr><th>zu</th><th>Form</th><th>Dauer bzw. Umfang</th><th>SL / QT</th></tr><tr><td>a)</td><td>Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte</td><td></td><td>SL</td></tr></table>	zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
zu	Form	Dauer bzw. Umfang	SL / QT						
a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL						
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung								
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.								
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).								
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch								
12	Modulbeauftragte/r: Prof. Dr. Johannes Blömer								

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Foundations of Cryptography:</i> Methodische Umsetzung Grundlegende Konzepte werden in einer Vorlesung präsentiert. Zusätzlich werden theoretische Konzepte in Tutorien in Kleingruppen vertieft. In schriftlichen Übungen und in Lesegruppen wird der praktischer Einsatz dieser Konzepte eingeübt. Lernmaterialien, Literaturangaben • Oded Goldreich, Foundations of Cryptography I,II, • Jonathan Katz, Yehuda Lindell, Introduction to Modern Cryptography • Folien der Vorlesung • Skript
----	--

3 Wahlpflichtmodule

Foundations of Knowledge Graphs							
Foundations of Knowledge Graphs							
Modulnummer: M.079.4054		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05806 Foundations of Knowledge Graphs	V2 Ü3	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Foundations of Knowledge Graphs:</i> Empfohlene Vorkenntnisse Kenntnisse in Graphentheorie und Logik sind hilfreich.						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Foundations of Knowledge Graphs:</i> Wissensgraphen werden in immer mehr Anwendungen eingesetzt. Große Organisationen wie Google, Yahoo! und die BBC nutzen diese Technologien, um den Zugriff auf die von ihnen verwalteten großen Datenmengen zu organisieren und zu verwalten. Diese Vorlesung soll Ansätze zum Aufbau, zur Speicherung, zur Abfrage und zur Nutzung von Wissensgraphen, insbesondere in intelligenten Anwendungen, vorstellen. Wir beginnen mit der Untersuchung von Techniken zur Wissensrepräsentation für Wissensgraphen. Anschließend werden Technologien zur Abfrage und Speicherung von Wissen (z. B. Graphdatenbanken) vorgestellt. Es folgen Techniken des maschinellen Lernens zum Einbetten von Wissensgraphen und deren Anpassung an Anwendungen, die auf klassischem maschinellem Lernen basieren. Abschließend werden wir spezielle Ansätze für das Lernen auf Wissensgraphen untersuchen. Die wichtigsten Themen des Moduls sind daher wie folgt: • Semantische Netzwerke • RDF(S)-Graphen • Property-Graphen • Beschreibungslogik und OWL • Einbettungen von Wissensgraphen • Erklärbares maschinelles Lernen auf Wissensgraphen						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden sind nach Absolvieren des Moduls in der Lage • Wissensgraphen einschließlich ihrer Semantik zu modellieren • Wissensgraphen abzufragen • Einbettungen für Wissensgraphen zu berechnen • Wissensgraphen in intelligenten Anwendungen zu verwenden			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch			
12	Modulbeauftragte/r: Prof. Dr. Axel-Cyrille Ngonga Ngomo			
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Foundations of Knowledge Graphs:</i> Methodische Umsetzung Wöchentliche Vorlesungen (2 SWS) mit neuen Inhalten zu dedizierten Themen. Prämissen und Umsetzungen dieser Prämissen werden vorgestellt. 1 SWS Seminar mit Übungsaufgaben zu den formalen und praktischen Konzepten aus der Vorlesung. 2 SWS Mini-Projekt zu einer komplexeren Aufgabe aus dem Themengebiet. Lernmaterialien, Literaturangaben Folien, Übungsaufgaben			

3 Wahlpflichtmodule

Geometric Deep Learning							
Geometric Deep Learning							
Modulnummer: M.079.4209		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05711 Geometric Deep Learning	V2 Ü3	75	105	P	50/25
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Geometric Deep Learning:</i> Empfohlene Vorkenntnisse Grundkenntnisse im Bereich Machine Learning können hilfreich sein. Erfahrung mit Python und PyTorch kann hilfreich sein.						

4	Inhalte: <i>Inhalte der Lehrveranstaltung Geometric Deep Learning:</i> Die Veranstaltung vermittelt Prinzipien und Techniken, die im Bereich des maschinellen Lernens, insbesondere mittels neuronaler Netze, von Bedeutung sind, wenn es um Eingabedaten, Ausgabedaten, oder Aufgabenstellungen geht, die dezidiert geometrischer Natur sind. Die besonderen Herausforderungen des Lernens auf riemannschen statt euklidischen Domänen werden erörtert. Der Aspekt der Invarianzen und Equivarianzen von Aufgabenstellungen wird betrachtet, gefolgt von Prinzipien diese bei der Konzeption oder beim Training neuronaler Netze explizit zu berücksichtigen. Ausgehend davon werden allgemeine Techniken und spezifische neuronale Architekturkomponenten erörtert, welche in Abhängigkeit der Datenrepräsentation geeignet sind, diese Prinzipien anzuwenden. Folgende Datenrepräsentationsformen werden betrachtet: • Point Clouds • Multi-View Images • Voxel Sets • Unfoldings • Graphs • Meshes • Distance Fields • Radiance Fields Folgende grundlegende Prinzipien, Aspekte und Techniken werden betrachtet: • Invarianzen und Equivarianzen • Scale Separation • Convolution • Pooling • Positional Encoding
5	Lernergebnisse und Kompetenzen: Die Studierenden können • grundlegende Komponenten neuronaler Architekturen für geometrische Daten reproduzieren und implementieren. • kontextabhängig neuronale Architekturen für wichtige Arten von Aufgabenstellungen unter Einbezug aufgabenspezifischen geometrischen Vorwissens auf Basis grundlegender Komponenten konzipieren. • grundlegende Repräsentationsformen geometrischer, insbesondere dreidimensionaler Daten erklären und hinsichtlich ihrer Eignung zur Eingabe- oder Ausgabekodierung im Rahmen des maschinellen Lernens vergleichen. • demonstrieren, dass das Ausnutzen geometrischer Strukturen und geometrischen Vorwissens im Kontext des maschinellen Lernens von hoher Bedeutung sein kann. • neuronale Architekturen auf ihre inhärenten Invarianzen und Equivarianzen hin untersuchen.

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme: keine		
8	Voraussetzungen für die Teilnahme an Prüfungen: keine		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Informatik v3		
12	Modulbeauftragte/r: Prof. Dr. Marcel Campen		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Geometric Deep Learning:</i> Methodische Umsetzung Inhalte werden im Rahmen der Vorlesung unter Nutzung von Präsentationsfolien und Tafelanschrieb vermittelt. In Präsenzübungen sowie in Heimarbeit werden die Inhalte vertieft und durch praktische Übungen ergänzt. Lernmaterialien, Literaturangaben • Vorlesungsfolien, Übungsblätter • Weitere Literatur wird in der Veranstaltung bekanntgegeben.		

3 Wahlpflichtmodule

Geometry Processing							
Geometry Processing							
Modulnummer: M.079.4205		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05803 Geometry Processing	V3 Ü2	75	105	P	60/30
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Geometry Processing:</i> Empfohlene Vorkenntnisse Grundkenntnisse in Programmierung, Linearer Algebra und Analysis. Erfahrung mit C++ kann hilfreich sein.						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Geometry Processing:</i> Die Veranstaltung vermittelt Konzepte, Techniken und Algorithmen der 3D-Geometriedatenverarbeitung. Ausgehend von der Betrachtung glatter Oberflächen mittels mathematischer Methoden erfolgt der Übergang zur Betrachtung diskreter Oberflächen mittels informatischer Methoden. Das Spektrum der verschiedenen digitalen Repräsentationsmöglichkeiten wird betrachtet, der Fokus auf die Repräsentation diskreter Oberflächen mittels Polygon- und Dreiecksnetzen gelegt. Die Verknüpfung von Einzelalgorithmen zu Pipelines wird anhand von Anwendungsfällen aus Bereichen wie Reverse Engineering und Digital Fabrication erörtert. Die Veranstaltung umfasst folgende Inhalte: • Netzerzeugung • Netzoptimierung • Netzvereinfachung • Netzglättung • Geometrische und topologische Formanalyse • Deformation und interaktive Modellierung • Oberflächenparametrisierung • Exaktes geometrisches Rechnen						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Nach Abschluss des Moduls können die Studierenden • parametrisch beschriebene Oberflächen auf ihre Krümmungsaspekte hin untersuchen • diskrete Oberflächen auf ihre Krümmungsaspekte hin untersuchen • grundlegende Algorithmen zur Netzerzeugung, -optimierung, -vereinfachung, -glättung und -parametrisierung reproduzieren und implementieren. • grundlegende Algorithmen zur Netzerzeugung, -optimierung, -vereinfachung, -glättung und -parametrisierung erklären und hinsichtlich ihrer jeweiligen Stärken und Schwächen vergleichen. • demonstrieren, dass solche Algorithmen in der Praxis meist in Form von Geometrieverarbeitungspipelines zum Einsatz kommen, und diese analysieren. • kontextabhängig Geometrieverarbeitungspipelines auf Basis grundlegender Algorithmen konzipieren. • geometrische Operationen und Algorithmen auf ihre numerischen Risiken hin untersuchen			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Informatik v3			
12	Modulbeauftragte/r: Prof. Dr. Marcel Campen			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Geometry Processing:</i> Methodische Umsetzung Inhalte werden im Rahmen der Vorlesung unter Nutzung von Präsentationsfolien und Tafelanschrieb vermittelt. In Präsenzübungen sowie in Heimarbeit werden die Inhalte vertieft und durch praktische Übungen ergänzt. Lernmaterialien, Literaturangaben • Vorlesungsfolien, Übungsblätter. • Weitere Literatur wird in der Veranstaltung bekanntgegeben.
----	---

3 Wahlpflichtmodule

Hardware Security							
Hardware Security							
Modulnummer: M.079.4211		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	K.079.42111 Hardware Security	V3 Ü2	75	105	P	30
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Hardware Security:</i> Basiskenntnisse in Programmierung, IT-Sicherheit und Kryptographie						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Hardware Security:</i> Die gesamte Software, die wir nutzen und auf die wir uns verlassen, wird letztendlich auf tatsächlicher Hardware ausgeführt. Die Interaktionen zwischen der Software und der Hardware, auf der sie ausgeführt wird, werden immer komplexer. Daher kann sich sichere Software als unsicher erweisen, wenn sie auf bestimmter Hardware ausgeführt wird. Angreifer können Implementierungsfehler, Logikfehler oder Seitenkanäle wie Timing oder Cache-Zugriffe ausnutzen, um Anwendungen zu knacken oder Geheimnisse aus kryptographischen Implementierungen zu extrahieren. Diese Vorlesung behandelt verschiedene Themen zur Sicherung und Nutzung sicherer Hardware. Der Kurs umfasst die folgenden Inhalte: • Kurzer Hintergrund in Kryptographie • Sichere Programmierung und Implementierungsfehler • Eingebettete und sichere Hardware • Smartcards, U2F-Token • Timing- und mikroarchitektonische Angriffe • Passive und aktive Seitenkanalangriffe • Abwehrmaßnahmen gegen Seitenkanalangriffe • Trusted Platform Modules, Secure Boot • Trusted Execution Environments • Hardware Security Module						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können: • Häufige Software- und Hardware-Schwachstellen identifizieren, die sichere Systeme gefährden, einschließlich Speicherfehlern und API-Missbrauch. • Die Grundideen hinter Seitenkanal- und Timing-Angriffen erklären. • Allgemeine Ansätze zum Schutz von Implementierungen vor diesen Angriffen identifizieren. • Die Architektur, Bedrohungsmodelle und Sicherheitseigenschaften von vertrauenswürdigen Hardwarekomponenten wie TPMs, TEEs und Smartcards beschreiben.											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><td>zu</td><td>Prüfungsform</td><td>Dauer bzw. Umfang</td><td>Gewichtung für die Modulnote</td></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: <table><tr><td>zu</td><td>Form</td><td>Dauer bzw. Umfang</td><td>SL / QT</td></tr><tr><td>a)</td><td>Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte</td><td></td><td></td></tr></table>				zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		
zu	Form	Dauer bzw. Umfang	SL / QT									
a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte											
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v3 (CEMA v3), Masterstudiengang Computer Engineering v3 (CEMA v3), englisch, Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Informatik v4											
12	Modulbeauftragte/r: Jan Jancar											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Hardware Security:</i> Methodische Umsetzung Die Themen werden durch Vorlesungspräsentationen vermittelt, wobei die Beteiligung der Studierenden und Fragen erwartet werden. Tutorien ergänzen dies um praktische Aufgaben zum Material. Schließlich erfordern Hausaufgaben Einzelarbeit und das Selbststudium von zusätzlichem Material. Lernmaterialien, Literaturangaben • Vorlesungs- und Tutoriumsfolien • The Hardware Hacking Handbook (ISBN: 9781593278748) und die dazugehörigen Notebooks auf hardwarehacking.io • Power Analysis Attacks: Revealing the Secrets of Smart Cards (ISBN 9780387308579) • Embedded Cryptography 1 (ISBN: 9781394351879) • Embedded Cryptography 2 (ISBN: 9781394351909) • Embedded Cryptography 3 (ISBN: 9781394351930)
----	---

3 Wahlpflichtmodule

Human Factors in Security and Privacy							
Human Factors in Security and Privacy							
Modulnummer: M.079.4092		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05733 Human Factors in Security and Privacy	V2 Ü3	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Human Factors in Security and Privacy:</i> Menschen sind wichtige Akteure im Bereich Sicherheit. Ein nachweislich sicheres System ist nur dann nützlich, wenn es auch tatsächlich von den Benutzern verwendet werden kann, und Systementwickler müssen menschliches Verhalten berücksichtigen, wenn sie sowohl Sicherheit als auch Benutzerfreundlichkeit gewährleisten wollen. In diesem Kurs untersuchen wir Faktoren der Benutzerfreundlichkeit von Sicherheit und Datenschutz anhand einer forschungsbasierten, projektorientierten Untersuchung. Wir behandeln Kernbereiche der Sicherheit und Privatsphäre sowie Methoden der Mensch-Computer-Interaktion (HCI), mit denen die Benutzerfreundlichkeit von Sicherheit und Privatsphäre gemessen werden kann. Von den Studierenden wird erwartet, dass sie Aufgaben zu diesem Thema bearbeiten und ein forschungsbasiertes Projekt abschließen. Die Veranstaltung umfasst folgende Inhalte: • Wie schreibt man eine wissenschaftliche Forschungsarbeit? • Wie führt man eine unabhängige wissenschaftliche Studie im Bereich Human Factors in Sicherheit und Datenschutz durch? • Methodik: qualitative, quantitative und „gemischte“ Methoden. • Einführung in die Forschung und wissenschaftliche Ethik • Einführung in die Literaturrecherche • Präsentation wissenschaftlicher Ergebnisse auf einer Konferenz						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden sind in der Lage • Forschungsmethoden im Bereich der menschlichen Faktoren in Bezug auf nutzbare Sicherheit und Privatsphäre zu verstehen und anzuwenden. • relevante Hypothesen und Forschungsfragen im Bereich der nutzbaren Sicherheit und Privatsphäre zu entwickeln. • eine Forschungsstudie zu entwerfen, durchzuführen und die Ergebnisse zu analysieren. • Ergebnisse unter Verwendung der besten Praktiken des wissenschaftlichen Schreibens zu beschreiben, zu untermauern und effektiv zu argumentieren. • ethische Fragen im Zusammenhang mit der Forschung zu menschlichen Faktoren im Bereich Sicherheit und Datenschutz zu verstehen. • die wichtigsten Themen und Schwerpunkte der nutzbaren Sicherheit und des Datenschutzes zu verstehen. • Forschungsergebnisse in der Veranstaltung zu präsentieren.											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><td>zu</td><td>Prüfungsform</td><td>Dauer bzw. Umfang</td><td>Gewichtung für die Modulnote</td></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: <table><tr><td>zu</td><td>Form</td><td>Dauer bzw. Umfang</td><td>SL / QT</td></tr><tr><td>a)</td><td>Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte</td><td></td><td>SL</td></tr></table>				zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
zu	Form	Dauer bzw. Umfang	SL / QT									
a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL									
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch											
12	Modulbeauftragte/r: Prof. Dr. Yasemin Acar											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Human Factors in Security and Privacy:</i> Methodische Umsetzung Die Inhalte werden in der Vorlesung vorgestellt und erläutert. Im begleitenden Tutorium werden die Vorlesungsthemen sowohl im Plenum als auch in Kleingruppen vertieft und diskutiert. Darüber hinaus präsentieren und diskutieren die Studierenden ihre Fortschritte bei den Projekten. Lernmaterialien, Literaturangaben • Aktuelle zugriffsfreie Forschungspapiere werden in der Veranstaltung bereitgestellt. • Redmiles, Elissa M., Yasemin Acar, Sascha Fahl, and Michelle L. Mazurek. A summary of survey methodology best practices for security and privacy researchers. 2017. https://drum.lib.umd.edu/bitstream/handle/1903/19227/CS-TR-5055.pdf • Weitere Literatur wird in der Veranstaltung bekanntgegeben.
----	---

3 Wahlpflichtmodule

Introduction to Description Logics							
Introduction to Description Logics							
Modulnummer: M.079.4098		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05832 Introduction to Description Logics	V3 Ü2	75	105	P	50/25
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Introduction to Description Logics:</i> Empfohlene Vorkenntnisse Kenntnisse aus den Modulen <i>Modellierung, Berechenbarkeit und Komplexität</i> und <i>Komplexitätstheorie</i> sind hilfreich – insbesondere zu Prädikatenlogik und den grundlegenden Komplexitätsklassen.						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Introduction to Description Logics:</i> In dieser Vorlesung mit Übung werden Beschreibungslogiken (BL) eingeführt. Im Detail werden die folgenden Inhalte behandelt: • Einführung der Beschreibungslogik <i>ALC</i> und Konzept- und Rollenoperatoren zur Erweiterung von <i>ALC</i>. • Es werden BL Wissensbasen besprochen und grundlegende Schlussfolgerungsprobleme für BLs eingeführt. • Beziehung von <i>ALC</i> zu Prädikatenlogik und Modallogik wird besprochen • Modelltheorie von <i>ALC</i> • Einführung Tableau-Algorithmus für Erfüllbarkeitstests in <i>ALC</i> • Komplexitätsanalyse von Schlussfolgern in <i>ALC</i> • Einführung von Schlussfolgerungsverfahren für Varianten der BL <i>EL</i> • Verfahren zur Anfragebeantwortung in BL						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können nach der Lehrveranstaltung: • Begriffe aus dem Gebiet der Beschreibungslogik fachlich sicher verwenden und erklären • Kennen Syntax und Semantik zentraler Elemente von Beschreibungslogik Wissensbasen sowie grundlegende Schlussfolgerungsprobleme für Beschreibungslogiken • Algorithmen zur Entscheidung von Schlussfolgerungsproblemen anwenden und im Hinblick auf die Berechnungskomplexität für verschiedene Beschreibungslogiken differenzieren • Beweismethoden für Beschreibungslogiken eigenständig nachvollziehen und auf kleine Instanzen anwenden.			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur oder mündliche Prüfung	90-120 min oder 30-45 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Informatik v3			
12	Modulbeauftragte/r: Prof. Dr.-Ing. Anni-Yasmin Turhan			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Introduction to Description Logics:</i> Methodische Umsetzung Diese Vorlesung orientiert sich stark an dem Lehrbuch "An Introduction to Description Logic". Die Vorlesung wird hauptsächlich als Folienvortrag gehalten. Beweise werden als Tafelanschrieb präsentiert oder als "flipped class room" behandelt. Die begleitenden Übungen vertiefen und ergänzen den in der Vorlesung präsentierten Stoff. Lehrmaterial und Literatur • "An Introduction to Description Logic" von Franz Baader, Ian Horrocks, Carsten Lutz, Uli Sattler (Dies ist als E-Book aus dem Netz der Universität Paderborn frei verfügbar.) • Weitere Literatur wird in der Veranstaltung bekanntgegeben.
----	---

3 Wahlpflichtmodule

Introduction to Quantum Computation							
Introduction to Quantum Computation							
Modulnummer: M.079.4059		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05706 Introduction to Quantum Computation	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Introduction to Quantum Computation:</i> Empfohlene Vorkenntnisse Lineare Algebra, Algorithmen						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Introduction to Quantum Computation:</i> In dieser Vorlesung werden die Grundlagen von Quanteninformatik und Quanteninformation vorgestellt. Das umfasst eine Einführung in Quantenmechanik, Quantenverschränkung, Quantenalgorithmen, Quantenfehlerkorrektur und Quanteninformation. • Quantenmechanik • Quantenverschränkung • Quantenalgorithmen • Quantenfehlerkorrektur • Quanteninformation						
5	Lernergebnisse und Kompetenzen: Studierende können: • die Postulate von Quantenmechanik beschreiben und benutzen, • die Benutzung von Quantenverschränkung als eine Quelle verstehen, • grundlegenden Quantenalgorithmen entwickeln und analysieren • Quantenfehlerkorrektur benutzen, • grundlegender Quanteninformationskonzepten, wie Entropie, verstehen und benutzen.						

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL / QT
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch		
12	Modulbeauftragte/r: Prof. Dr. Sevag Gharibian		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Introduction to Quantum Computation:</i> Methodische Umsetzung Eine Mischung aus Folien und Tafelanschrieb. Alle wichtigen Konzepte und Techniken werden in Übungen anhand von Beispielen weiter vertieft. Lernmaterialien, Literaturangaben • Michael A. Nielsen, Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press • Vorlesungsfolien, Übungsaufgaben		

3 Wahlpflichtmodule

Logic Programming for Artificial Intelligence																					
Logic Programming for Artificial Intelligence																					
Modulnummer: M.079.4031		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester															
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en															
1	Modulstruktur <table><tr><td></td><td>Lehrveranstaltung</td><td>Lehrform</td><td>Kontaktzeit (h)</td><td>Selbststudium (h)</td><td>Status (P/WP)</td><td>Gruppengröße (TN)</td></tr><tr><td>a)</td><td>L.079.05808 Logic Programming for Artificial Intelligence</td><td>V3 Ü2</td><td>75</td><td>105</td><td>P</td><td>70/35</td></tr></table>								Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)	a)	L.079.05808 Logic Programming for Artificial Intelligence	V3 Ü2	75	105	P	70/35
	Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)															
a)	L.079.05808 Logic Programming for Artificial Intelligence	V3 Ü2	75	105	P	70/35															
2	Wahlmöglichkeiten innerhalb des Moduls: keine																				
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Logic Programming for Artificial Intelligence:</i> Empfohlene Vorkenntnisse Kenntnisse in der Programmierung werden in dem Umfang vorausgesetzt, wie sie in den Veranstaltungen „Einführung in die Programmierung“ und „Programmiersprachen“ gelehrt werden, sowie Kenntnisse in Datenbank-Anfragesprachen, wie sie in der Veranstaltung „Datenbanksysteme“ gelehrt werden.																				
4	Inhalte: <i>Inhalte der Lehrveranstaltung Logic Programming for Artificial Intelligence:</i> Die Lehrveranstaltung behandelt Methoden der symbolischen KI und deren Implementierung durch Logik-Programmierung. Ein Fokus liegt dabei auf der Entwicklung kleiner Demo-Programme, die die Grundideen verschiedener Methoden aus der KI zeigen. Die Veranstaltung umfasst folgende Inhalte: • Constraint Logic Programming und logische Puzzles, • Suche und Strategiespiele, • Parsing natürlicher Sprache und Semantik-Konstruktion, • Sprachdialogsysteme, • automatisches Übersetzen, • Spracherweiterung und Interpreterbau, • Termersetzungssysteme, Logik und Erklärungssysteme																				

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Nach Abschluss dieses Moduls können Studierende • Programme entwickeln, die logische Puzzles durch Constraint Logic Programming lösen • Methoden zur Graphsuche und ihre Eignung in Strategiespielen bewerten • die grundlegenden syntaktischen und semantischen Konzepte beim Parsen von Texten und ihre Verwendung in Sprachdialogsystemen und automatischer Übersetzung erklären. • die Chancen und Herausforderungen sowie den Aufwand von Spracherweiterungen bewerten • eigene Kalküle entwerfen und in denen programmieren • Inferenzsysteme um Erklärungskomponenten erweitern und die Erklärungsfähigkeit von KI-Systemen bewerten											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><td>zu</td><td>Prüfungsform</td><td>Dauer bzw. Umfang</td><td>Gewichtung für die Modulnote</td></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>120-180 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: <table><tr><td>zu</td><td>Form</td><td>Dauer bzw. Umfang</td><td>SL / QT</td></tr><tr><td>a)</td><td>Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte</td><td></td><td>SL</td></tr></table>				zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
zu	Form	Dauer bzw. Umfang	SL / QT									
a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL									
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: keine											
12	Modulbeauftragte/r: Prof. Dr. Stefan Böttcher											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Logic Programming for Artificial Intelligence:</i> Methodische Umsetzung Die Inhalte werden in einer Präsentation im Rahmen einer Vorlesung eingeführt und durch Demoprogramme ergänzt. Anschließend werden die Inhalte in Präsenzübungen in Kleingruppen sowie in Heimübungen vertieft und durch praktische Übungen ergänzt. Bei einigen Programmierübungen können automatische Feedback-Systeme das selbständige Lernen der Studierenden unterstützen. Lernmaterialien, Literaturangaben • Iwan Bratko: Logic Programming for Artificial Intelligence, neueste Ausgabe. • Sterling/Shapiro: The Art of Prolog, neueste Ausgabe. • Peter Flach: Simply Logical. https://book.simply-logical.space/src/simply-logical.html
----	---

3 Wahlpflichtmodule

Machine Learning 1							
Machine Learning 1							
Modulnummer: M.079.4032		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
	Lehrveranstaltung		Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05701 Machine Learning 1	V3 Ü2	75	105	P	75/25
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Machine Learning 1:</i> Empfohlene Vorkenntnisse Grundlagen in Mathematik (lineare Algebra, Statistik), Programmierung und Algorithmen.						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Machine Learning 1:</i> Aufgrund der stetig wachsenden Menge an Daten, die in unserer Informationsgesellschaft systematisch produziert wird, hat das Maschinelle Lernen in den letzten Jahren mehr und mehr an Bedeutung gewonnen, nicht nur als wissenschaftliche Disziplin sondern auch als Schlüsseltechnologie für moderne Software und intelligente Systeme. Diese Vorlesung gibt eine Einführung in das Maschinelle Lernen, wobei der Fokus auf dem überwachten Lernen für Klassifikation und Regression liegt. Theoretische Grundlagen der Generalisierung werden ebenso behandelt wie praktische Aspekte und konkrete Lernalgorithmen. • Introduction • Foundations (The learning problem, Generalization Theory, Bias-Variance tradeoff) • Techniques (e.g., The linear model, non-linear techniques, SVM, tree-based methods, ensembles, deep learning) • Validation and practical implementations (metrics, training vs testing, cross-validation, AutoML)						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden • verstehen die statistischen Grundlagen der Generalisierung, d.h. der Induktion von Modellen aus Daten, sowie den gesamten Prozess des maschinellen Lernens. • können grundlegende Methoden und Algorithmen des überwachten Lernens für Klassifizierung und Regression implementieren und in der Praxis anwenden. Gegebenenfalls können sie auch notwendige Anpassungen vornehmen. • verstehen praktische Tools für die Modellvalidierung und für das automatisierte maschinelle Lernen und können diese anwenden. • bewerten die Vor- und Nachteile des Einsatzes verschiedener Techniken des maschinellen Lernen in verschiedenen Situationen.			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Credits gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Informatik v3			
12	Modulbeauftragte/r: Prof. Dr. Heike Trautmann			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Machine Learning 1:</i> Methodische Umsetzung Theoretische Grundlagen und Konzepte des Maschinellen Lernens werden im Rahmen einer Vorlesung eingeführt und anschließend in praktischen Übungen in Kleingruppen sowie in Heimübungen vertieft ergänzt. Lernmaterialien, Literaturangaben • Skript • Y.S. Abu-Mostafa, M. Magdon-Ismael, H.T. Lin. Learning from Data, AMLBook, 2012. • P. Flach. Machine Learning, Cambridge Univ. Press, 2012. • E. Alpaydin. Machine Learning, Oldenbourg, 2008. • C.M. Bishop. Pattern Recognition and Machine Learning, Springer, 2006.
----	--

3 Wahlpflichtmodule

Model-Based Systems Engineering							
Model-Based Systems Engineering							
Modulnummer: M.079.4062		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
	Lehrveranstaltung		Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05810 Model-Based Systems Engineering	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Model-Based Systems Engineering:</i> Empfohlene Vorkenntnisse Grundlagen des Systems Engineerings						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Model-Based Systems Engineering:</i> Durch den technischen Wandel von mechatronischen zu intelligenten technischen Systemen (ITS) stehen Unternehmen und Entwicklungsteams vor vielen Herausforderungen. Wesentlich ist die Zunahme der Komplexität und Vernetzung von Systemen (Produkte). Bestehende Ansätze in der Produktentstehung können diese nicht effizient und effektiv abdecken. Model-based Systems Engineering (MBSE) stellt sich hierfür als vielversprechender Ansatz auf, die Herausforderungen zu lösen. MBSE sieht sich als Weiterentwicklung des Systems Engineerings und baut auf dessen Grundlagen auf. Dabei wird das Systems Engineering, welches primär auf Dokumenten basiert durch die Einführung von Modellen erweitert. Die Veranstaltung umfasst folgende Inhalte: • Intelligente technische Systeme • Model-based Systems Engineering 101 • Systems Modelling Grundlagen • Sprachen und Methoden - CONSENS, SysML • Systems Architecting • IT-Tools für MBSE						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden • erwerben durch die Veranstaltung ein solides Verständnis über Model-Based System Engineering, • kennen verschiedene Methoden, Sprachen und Werkzeuge, • können das erlangte Wissen anwenden • können eigenständig Lösungen erarbeiten und gegenüber den Dozenten kommunizieren.		
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme: keine		
8	Voraussetzungen für die Teilnahme an Prüfungen: keine		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Master's Program Electrical Systems Engineering (ESEMA v2), Master's Program Electrical Systems Engineering v3 (ESEMA v3)		
12	Modulbeauftragte/r: Prof. Dr.-Ing. Roman Dumitrescu		

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Model-Based Systems Engineering:</i> Methodische Umsetzung Das Modul besteht aus zwei Teilen 1. Vorlesung mit Folien (Lecture): Grundlagen und Konzepte werden in der Vorlesung erklärt und anhand von Beispielen veranschaulicht. 2. Übungen (Tutorial): In der Übung erfolgt ein Wissenstransfer und Anwendung der Konzepte. Die Übungen sind in Eigenarbeit vorzubereiten. Lernmaterialien, Literaturangaben • Gausemeier, J.; Dumitrescu, R.; Steffen, D.; Czaja, A.; Wiederkehr, O.; Tschirner, C.: Systems Engineering in industrial practice. Heinz Nixdorf Institute, University Paderborn, 2013, Unter: https://www.hni.uni-paderborn.de/en/spe/systemsengineering/ • Dumitrescu, R.; Albers, A.; Riedel, O.; Stark, R.; Gausemeier, J. (Eds): Engineering in Germany – Status quo in Business and Science. Federal Ministry of Education and Research, 2021 Unter: https://www.advanced-systems-engineering.de/#studie • Weitere Literatur wird in der Veranstaltung bekanntgegeben.
----	---

3 Wahlpflichtmodule

Multi-Objective Optimisation																					
Multi-Objective Optimisation																					
Modulnummer: M.079.4095		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester															
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en															
1	Modulstruktur <table><tr><td></td><td>Lehrveranstaltung</td><td>Lehrform</td><td>Kontaktzeit (h)</td><td>Selbststudium (h)</td><td>Status (P/WP)</td><td>Gruppengröße (TN)</td></tr><tr><td>a)</td><td>L.079.05821 Multi-Objective Optimisation</td><td>V3 Ü2</td><td>75</td><td>105</td><td>P</td><td>30/15</td></tr></table>								Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)	a)	L.079.05821 Multi-Objective Optimisation	V3 Ü2	75	105	P	30/15
	Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)															
a)	L.079.05821 Multi-Objective Optimisation	V3 Ü2	75	105	P	30/15															
2	Wahlmöglichkeiten innerhalb des Moduls: keine																				
3	Teilnahmevoraussetzungen: <i>Teilnahmevoraussetzungen der Lehrveranstaltung Multi-Objective Optimisation:</i> Empfohlene Vorkenntnisse Solide Grundkenntnisse in Algorithmen und Datenstrukturen, Mathematik, und Grundkenntnisse im Bereich Optimierung sind hilfreich.																				
4	Inhalte: <i>Inhalte der Lehrveranstaltung Multi-Objective Optimisation:</i> Optimierungsprobleme sind allgegenwärtig und wir alle lösen sie (näherungsweise) im Alltag etwa bei der Routenfindung mit Google Maps um schnell von A nach B zu kommen oder der Entscheidung für eine Kasse mit der kürzesten Warteschlange (kürzester erwarteter Wartezeit) im Supermarkt. Optimierungsprobleme sind jedoch selten einkriteriell. Vielmehr sind sie mehrkriterieller Natur und die einzelnen Ziele stehen üblicherweise miteinander in Konflikt. So kann bei der Routenplanung einerseits die zurückgelegte Strecke von Relevanz sein (je kürzer desto besser) und zum anderen der Treibstoffverbrauch (je geringer desto besser); der kürzeste Weg kann etwa durch die Innenstadt führen mit vielen Stop-and-Go Manövern durch rote Ampeln vor allem zu Stoßzeiten. Ein Weg um die Innenstadt herum hingegen kann trotz längerer Strecke weniger Treibstoff verbrauchen. Entsprechend ist das Ziel in der Mehrziel-Optimierung das Auffinden einer Menge von optimalen Kompromisslösungen. Dieser Kurs gibt eine umfassende Einführung in mehrkriterielle Optimierung (multi-objective optimisation) und die damit einhergehenden Herausforderungen. Neben klassischen allgemeinen Ansätzen werden exakte Verfahren für ausgewählte kombinatorische Optimierungsprobleme sowie heuristische (natur-inspirierte) Verfahren vorgestellt. Weiterhin behandelt der Kurs heuristische Lösungsansätze für Probleme mit mehr als drei Kriterien (many-objective optimisation).																				

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können • wichtige exakte Algorithmen für mehrkriterielle minimale Spannbaum Problem und das mehrkriterielle kürzeste Wege Problem erklären, implementieren und anwenden • die Grenzen exakter Algorithmen für mehrkriterielle Probleme verstehen • biologisch inspirierte Heuristiken für Mehrzielprobleme erklären und anwenden • die Qualität der berechneten Ergebnisse mehrkriterieller Algorithmen beurteilen, evaluieren und visualisieren • Herausforderungen von Problemen mit mehr als drei Kriterien verstehen und Lösungsansätze erklären											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: <table><tr><th>zu</th><th>Form</th><th>Dauer bzw. Umfang</th><th>SL / QT</th></tr><tr><td>a)</td><td>Übungsaufgaben</td><td></td><td>SL</td></tr></table>				zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben		SL
zu	Form	Dauer bzw. Umfang	SL / QT									
a)	Übungsaufgaben		SL									
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Informatik v3											
12	Modulbeauftragte/r: Prof. Dr. Heike Trautmann											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Multi-Objective Optimisation:</i> Methodische Umsetzung Folien-gestützte Vorlesung mit eingestreuten Übungsaufgaben. In der Übung erfolgt der Wissenstransfer und die Anwendung des Erlernten sowohl in theoretischen als auch praktischen Übungsaufgaben. Lernmaterialien, Literaturangaben • Deb, Kalyanmoy. „Multi-Objective Optimization Using Evolutionary Algorithms“. • Ehrgott, Matthias. Multicriteria Optimization. Bd. 491. Lecture Notes in Economics and Mathematical Systems. Berlin, Heidelberg: Springer, 2000. • Weitere Literatur wird in der Veranstaltung bekanntgegeben.
----	--

3 Wahlpflichtmodule

Post-Quantum Cryptography							
Post-Quantum Cryptography							
Modulnummer: M.079.4089		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
	Lehrveranstaltung		Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
a)	L.079.05732 Post-Quantum Cryptography		V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine Teilnahmevoraussetzungen der Lehrveranstaltung Post-Quantum Cryptography: Empfohlene Vorkenntnisse Grundkenntnisse in Kryptographie und Komplexitätstheorie						
4	Inhalte: Inhalte der Lehrveranstaltung Post-Quantum Cryptography: IT-Sicherheit beruht zu großen Teilen auf Verfahren der modernen Kryptographie. Hierzu gehören viele Verfahren der so genannten Public-Key Kryptographie wie das RSA- und das Elgamal-Verschlüsselungsverfahren, das RSA-Unterschriftenverfahren sowie die verschiedenen Varianten des Digital Signature Algorithms (DSA). Im Jahr 1994 stellte Peter Shor einen effizienten Algorithmus zur Berechnung der Primfaktorisation ganzer Zahlen und zur Berechnung diskreter Logarithmus in endlichen Gruppen vor. Damit sind alle genannten Verfahren der Public-Key Kryptographie unsicher, wenn Quantencomputer hinreichender Größe und Komplexität realisiert werden können. Es ist daher wichtig, Alternativen zu klassischen Public-Key Verfahren zu entwickeln, die zumindest nach aktuellem Forschungsstand nicht von Quantencomputern gebrochen werden können. Wichtige Kandidaten (und teilweise kurz vor der Standardisierung stehende) für solche post-quanten sichere Verfahren beruhen auf Techniken fehler-korrigierender Codes und der Geometrie der Zahlen. In dieser Vorlesung sollen wichtige Kandidaten für post-quanten sichere Verfahren vorgestellt und diskutiert werden. Die Veranstaltung umfasst folgende Inhalte: • Einführung in Codes, Gitter, diskretisierte Gaußverteilungen • gitter- und codebasierte Verschlüsselung • gitterbasierte Signaturen • Gitter und Zero-Knowledge Beweise • gitterbasierte Gruppensignaturen						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können • den Unterschied zwischen klassischer und post-quanten Sicherheit verstehen und erklären. • die Bedeutung von post-quanten Kryptographie für ausgewählte Anwendungen erklären. • Konzepte aus dem Bereich der Geometrie der Zahlen und der fehler-korrigierende Codes erklären und anwenden. • wichtige Konstruktionen aus dem Bereich der post-quanten Kryptographie erklären und deren Sicherheit beweisen. • Sicherheitsannahmen aus der post-quanten Kryptographie erläutern und gegebenenfalls für neue post-quanten Primitiven einsetzen.			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch			
12	Modulbeauftragte/r: Prof. Dr. Johannes Blömer			

3 Wahlpflichtmodule

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Post-Quantum Cryptography:</i> Methodische Umsetzung Grundlegende Konzepte werden in einer Vorlesung präsentiert. Zusätzlich werden theoretische Konzepte in Tutorien in Kleingruppen sowie in schriftlichen Übungen vertieft. Lernmaterialien, Literaturangaben Verweise auf aktuelles Lernmaterial werden in der Vorlesung gegeben.
----	---

3 Wahlpflichtmodule

Privacy and Technology							
Privacy and Technology							
Modulnummer: M.079.4087		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05705 Privacy and Technology	V2 Ü3	75	105	WP	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine						
4	Inhalte: Inhalte der Lehrveranstaltung Privacy and Technology: Dieser Kurs vermittelt den Studierenden ein grundlegendes Verständnis der Risiken und Grundsätze für Privatsphäre und Datenschutz, der gängigsten Technologien zur Bewältigung und wie menschliche Faktoren die Gestaltung beeinflussen. Der Kurs analysiert darüber hinaus Angreifermodelle und Bewertungsmetriken, die der Entwicklung von Technologien zur Verbesserung der Privatsphäre zugrunde liegen. Darüber hinaus wird ein kurzer Überblick über die nutzbare Sicherheit sowie das Identitätsmanagement und entsprechende Fallstudien gegeben. Hierfür ist ein oberflächliches Wissen über HCI-Grundlagen wünschenswert. Durch die Sichtung relevanter Paper und das Halten von Präsentationen werden die Studierenden mit dem neuesten Stand der Forschung auf diesem Gebiet vertraut gemacht und lernen, wie man wissenschaftlich arbeitet. Die Veranstaltung umfasst folgende Inhalte: • Metriken zum Schutz der Privatsphäre und Angreifermodelle • Anonyme Kommunikation • Technologien zur Verbesserung der Privatsphäre • Anonymisierungsalgorithmen für Datenbanken • Homomorphe Verschlüsselung und Null-Wissen-Beweise • Datenschutz beim Identitätsmanagement • Menschlicher Faktor beim Thema Privatsphäre & Datenschutz • Anwendung von Datenschutzprinzipien und Fallstudien						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden • sind in der Lage, über Datenschutz und Privatsphäre kritisch zu urteilen, • erwerben Kenntnisse über die Beurteilung von Datenschutzrisiken, • entwickeln ein Verständnis über Gestaltungsaspekte von Technologien, die zur Verbesserung der Privatsphäre führen, • lernen den aktuellen Forschungsstand zum Thema kennen und • analysieren und diskutieren Lösungen zu einem gegebenen Datenschutzproblem			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch			
12	Modulbeauftragte/r: Prof. Dr. Patricia Arias Cabarcos			

3 Wahlpflichtmodule

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Privacy and Technology:</i> Methodische Umsetzung Die Inhalte werden durch eine Präsentation in Form einer Vorlesung vermittelt. Dazu werden sie in Präsenzübungen in Kleingruppen, sowie durch individuelle Präsentationen vertieft und Methoden durch praktische Übungen umgesetzt und angewandt. Lernmaterialien, Literaturangaben • Vorlesungsfolien, wissenschaftliche Literatur und spezifische Lektüre werden während des Kurses zur Verfügung gestellt.
----	--

3 Wahlpflichtmodule

Quantum Algorithms							
Quantum Algorithms							
Modulnummer: M.079.4072		Workload (h): 180	Leistungspunkte: 6		Turnus: Sommersemester		
		Studiensemester: 1-3	Dauer (in Sem.): 1		Sprache: en		
1	Modulstruktur						
	Lehrveranstaltung		Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
a)	L.079.05830 Quantum Algorithms		V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Quantum Algorithms:</i> Empfohlene Vorkenntnisse Lineare Algebra, Quanteninformatik						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Quantum Algorithms:</i> Die Vorlesung stellt fortgeschrittene Methoden vor, um Quantenalgorithmen zu entwickeln. Inhalt: • Quantenschaltung • Quantenalgorithmen für algebraische Problemen • Quantum Walks • Quanten Query Komplexität • Adiabatische Quantencomputing						
5	Lernergebnisse und Kompetenzen: Studierende können: • universelle Gatter beschreiben und benutzen, • die Quanten-Fourier-Transformation benutzen, • Quantum Walks benutzen, • adiabatische Quantenalgorithmen entwickeln, • mit Quanten-Query-Komplexität arbeiten						

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL / QT
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch		
12	Modulbeauftragte/r: Prof. Dr. Sevag Gharibian		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Quantum Algorithms:</i> Methodische Umsetzung Eine Mischung aus Folien und Tafelanschrieb. Alle wichtigen Konzepte und Techniken werden in Übungen anhand von Beispielen weiter vertieft. Lernmaterialien, Literaturangaben • Michael A. Nielsen, Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press • Andrew M. Childs, Wim van Dam, Quantum algorithms for algebraic problems, Reviews of Modern Physics, volume 82, 2010 • Vorlesungsfolien, Übungsaufgaben		

3 Wahlpflichtmodule

Quantum Complexity Theory							
Quantum Complexity Theory							
Modulnummer: M.079.4063		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05811 Quantum Complexity Theory	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Quantum Complexity Theory:</i> Empfohlene Vorkenntnisse Lineare Algebra, Quanteninformatik						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Quantum Complexity Theory:</i> Diese Vorlesung gibt einen kurzen Überblick über die Grundlagen von Quanteninformatik und wendet sich anschließend der Quantenkomplexitätstheorie zu. Dabei werden sowohl einführende als auch vertiefende Themen behandelt wie die Analoga zu P und NP (bezeichnet als BQP, QCMA, and QMA), Quanten-Erfüllbarkeitsprobleme, Quanten-interaktive Beweise und Tensor-Netzwerke. Begleitend wird semidefinite Programmierung als ein wichtiges Werkzeug eingeführt. • Komplexitätsklassen BQP, QCMA, QMA • Quantenalgorithmen für die Lösung linearer Systeme • Quanten-Erfüllbarkeitsprobleme • Quanten-interaktive Beweise • Semidefinite Programmierung						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden sind in der Lage • Sprachklassen von Versprechensklassen zu unterscheiden • grundlegende Quantenkomplexitätsklassen, wie BQP und QMA, zu definieren • BQP-Härte-Resultate über Polynomialzeit-Reduktionen zu beweisen • Beweis von QMA-Härteergebnissen über Polynomialzeit-Reduktionen • Anwendung der semidefiniten Programmierung zur Analyse von interaktiven Quantenbeweisen			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch			
12	Modulbeauftragte/r: Prof. Dr. Sevag Gharibian			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Quantum Complexity Theory:</i> Methodische Umsetzung Eine Mischung aus Folien und Tafelanschrieb. Alle wichtigen Konzepte und Techniken werden in Übungen anhand von Beispielen weiter vertieft. Lernmaterialien, Literaturangaben • Michael A. Nielsen, Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press • S. Gharibian, Y. Huang, Z. Landau, S. W. Shin, Quantum Hamiltonian Complexity, Foundations and Trends in Theoretical Computer Science • Vorlesungsfolien, Übungsaufgaben
----	---

3 Wahlpflichtmodule

Quantum Information							
Quantum Information							
Modulnummer: M.079.4090		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05805 Quantum Information	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Quantum Information:</i> Empfohlene Vorkenntnisse Lineare Algebra						

4	Inhalte: <i>Inhalte der Lehrveranstaltung Quantum Information:</i> Im Laufe des letzten Jahrhunderts hat die Quantenmechanik tiefgreifende Auswirkungen auf die Grundlagenwissenschaft und die Technologie gehabt. Das neu entstehende Gebiet der Quanteninformationstheorie untersucht ein Paradigma für die Informationsverarbeitung auf der Grundlage der Quantenmechanik. Dieser Bereich hat gezeigt, dass die Quanteninformationsverarbeitung ihr klassisches Gegenstück übertreffen kann und stellt eine revolutionäre Richtung für die Erforschung künftiger Informationstechnologien dar. Die Quanteninformationswissenschaft umfasst Techniken aus der Informatik, Mathematik und Physik. Von besonderem Interesse ist die Quantenverschränkung, ein Phänomen, das auftritt, wenn eine Gruppe von Teilchen so erzeugt wird oder miteinander interagiert, dass der Zustand jedes Teilchens nicht unabhängig von den anderen beschrieben werden kann, selbst wenn die Teilchen durch beliebig große Abstände getrennt sind. Die Verschränkung ist ein Hauptmerkmal der Quantenmechanik, das in der klassischen Physik nicht vorkommt, und sie ist eine Ressource hinter den meisten modernen Quantentechnologien, wie z. B. Quantencomputern. In dieser Vorlesung werden die fortgeschrittenen Konzepte der Quantenkommunikation und -information vorgestellt. Der Inhalt umfasst: • Verschränkung von Zwei- und Vielteilchensystemen • Quanteninformationsverarbeitung und Anwendungen • Maße für Verschränkung, Abstand und Treue • Höhere lokale Dimensionen (Qubits vs. Qudits) • Quantenkanäle • Klassische und Quanten-Fehlerkorrekturcodes und ihre Unterschiede
5	Lernergebnisse und Kompetenzen: Die Studierenden lernen modernste Konzepte an der Schnittstelle von Informatik und Quantenmechanik kennen. Die Vorlesung vermittelt den Studierenden fortgeschrittene, interdisziplinäre technische Kenntnisse, die sie in die Lage versetzen, Karrieren in analyseintensiven Industrien, Technologie-Start-ups oder Forschungs- und Entwicklungsaufgaben in führenden Technologieunternehmen oder in der Wissenschaft zu verfolgen. Um dies zu erreichen, werden die Studierenden mit den Grundlagen der Quantenmechanik und der zugehörigen Algebra vertraut gemacht. Darüber hinaus werden sie in der Lage sein: • die zugrundeliegenden Konzepte von verschränkten Systemen (Zweikörper- und Vielkörpersysteme) zu verstehen, • die Grundidee der maximal verschränkten Systeme zu verstehen, sie zu klassifizieren und für praktische Anwendungen zu charakterisieren, • den grundlegenden Begriff der Teilchen höherer lokaler Dimensionen beschreiben (Qubits vs. Qudits), • die Theorie der klassischen und der Quanten-Fehlerkorrekturcodes anwenden und ihre Unterschiede untersuchen • interdisziplinäre Themen zu bearbeiten und sich insbesondere die Grundlagen verschiedener Disziplinen anzueignen.

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL / QT
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch		
12	Modulbeauftragte/r: Dr. Zahra Raissi		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Quantum Information:</i> Methodische Umsetzung Theoretische Grundlagen und Konzepte werden in Form von Vorlesungen vermittelt und in praktischen Übungen, Gruppenarbeiten sowie individuellen Übungsaufgaben vertieft. Lernmaterialien, Literaturangaben • Michael A. Nielsen, Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2000. • F. J. MacWilliams and N. J. A. Sloane. The Theory of Error-Correcting Codes, North-Holland Mathematical Library. North-Holland, Amsterdam, 1977. ISBN 9780444851932. • Ingemar Bengtsson and Karol Zyczkowski, Geometry of quantum states: an introduction to quantum entanglement, Cambridge university press, 2006, ISBN 9780511535048. • Vorlesungsfolien • Übungsaufgaben		

3 Wahlpflichtmodule

Real-Time Operating Systems: Design and Implementations							
Real-Time Operating Systems: Design and Implementations							
Modulnummer: M.079.4213		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	Real-Time Operating Systems: Design and Implementations	V2 Ü3	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: <i>Teilnahmevoraussetzungen der Lehrveranstaltung Real-Time Operating Systems: Design and Implementations:</i> Empfohlene Vorkenntnisse Kenntnisse in den Bereichen Betriebssysteme, eingebettete Systeme und C/C++-Programmierung sind von Vorteil. Grundkenntnisse in Computerarchitektur und paralleler Programmierung sind hilfreich, aber nicht erforderlich.						

3 Wahlpflichtmodule

4	Inhalte: <i>Inhalte der Lehrveranstaltung Real-Time Operating Systems: Design and Implementations:</i> Echtzeitbetriebssysteme sind für eingebettete und cyber-physische Systeme unverzichtbar, in denen zeitnahe und vorhersehbare Reaktionen erforderlich sind. Dieses Modul vermittelt die Grundlagen des Entwurfs und der Implementierung von Echtzeitbetriebssystemen. Es verbindet konzeptionelle Grundlagen mit praktischen Übungen, in denen die Studierenden Mechanismen eines Echtzeitbetriebssystems auf einer eingebetteten Plattform analysieren, erweitern und bewerten. • Echtzeitsysteme und zeitliche Anforderungen • Architektur und grundlegende Dienste von Echtzeitbetriebssystemen • Task-Modelle, Task-Zustände und Task-Verwaltung • periodische Ausführung und Timer-Mechanismen • Interprozesskommunikation und Synchronisation • Interrupt-Behandlung und Interaktion zwischen Tasks und Interrupts • Kontextwechsel und Task-Ausführung • Scheduling mit fester und dynamischer Priorität • Implementierung von Scheduling-Mechanismen, z. B. EDF • gemeinsame Ressourcennutzung und Synchronisationsprotokolle, z. B. PIP und PCP • serverbasierte Scheduling- und Reservierungsmechanismen • praktische Implementierung und Bewertung von RTOS-Mechanismen auf einer Embedded-Plattform								
5	Lernergebnisse und Kompetenzen: Die Studierenden können • die Rolle, den Aufbau und die Dienste von Echtzeitbetriebssystemen in eingebetteten und cyber-physischen Systemen erläutern; • Zeitanforderungen, Aufgabenmodelle, Planungsentscheidungen und Situationen der gemeinsamen Ressourcennutzung in Echtzeitsystemen analysieren; • Konzepte der Echtzeitplanung und -synchronisation auf konkrete Systembeispiele anwenden; • ausgewählte RTOS-Mechanismen, wie z. B. Planungs- oder Protokolle zur gemeinsamen Ressourcennutzung, auf einer eingebetteten Plattform implementieren; • das Verhalten von RTOS-Mechanismen hinsichtlich Zeitsteuerung, Ressourcenmanagement und Vorhersagbarkeit bewerten; • Kompromisse beim Entwurf zwischen Funktionalität, Vorhersagbarkeit der Zeitsteuerung, Implementierungskomplexität und Systemrobustheit begründen.								
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote						
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%						

3 Wahlpflichtmodule

7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Informatik v4		
12	Modulbeauftragte/r: Prof. Dr.-Ing. Kuan-Hsun Chen		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Real-Time Operating Systems: Design and Implementations:</i> Methodische Umsetzung Das Modul verbindet Vorlesungen mit praktischen Übungen und labororientierten Aufgaben. In den Vorlesungen werden die konzeptionellen und analytischen Grundlagen von Echtzeitbetriebssystemen vermittelt. In den Übungen vertiefen die Studierenden diese Konzepte anhand von Aufgaben und implementieren ausgewählte RTOS-Mechanismen auf einer Embedded-Plattform. Die praktischen Arbeiten dienen dazu, theoretische Konzepte mit dem beobachtbaren Systemverhalten zu verknüpfen. Lernmaterialien, Literaturangaben • Giorgio C. Buttazzo, Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications, Springer. • Jane W. S. Liu, Real-Time Systems, Prentice Hall. • Hermann Kopetz, Real-Time Systems: Design Principles for Distributed Embedded Applications, Springer. • Für die praktischen Aufgaben werden die FreeRTOS-Dokumentation sowie ausgewählte technische Unterlagen herangezogen. • Weitere Literatur wird in der Veranstaltung bekanntgegeben.		

3 Wahlpflichtmodule

Real World Crypto Engineering							
Real World Crypto Engineering							
Modulnummer: M.079.4067		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05704 Real World Crypto Engineering	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Real World Crypto Engineering:</i> Empfohlene Vorkenntnisse Basiskonntnisse in Programmierung, IT-Sicherheit und Kryptographie						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Real World Crypto Engineering:</i> Starke Kryptographie ist nicht immer ausreichend, um die grundlegenden Sicherheitsziele zu schützen. Auch wenn starke kryptographische Algorithmen verwendet werden, kann bei deren Einsatz viel schief gehen. In dieser Vorlesung werden wir auf die wichtigsten Protokolle und kryptographische Schutzmechanismen eingehen (z.B. TLS, SSH, WPA) und werden ihre Basiskonzepte kennenlernen. Anschließend werden wir prominente Angriffe vorstellen, die die gewünschten Sicherheitsziele komplett gebrochen haben. Basierend auf vielen Fällen werden wir lernen, was beim Design und bei der Implementierung von kryptographischen Anwendungen wichtig ist. Die Veranstaltung umfasst folgende Inhalte: • Kurze Einführung in die Kryptographie • TLS (Transport Layer Security) • Angriffe auf TLS (z.B. ROBOT, DROWN, Invalid Curve) • Evaluation von Implementierungen mit systematischen Methoden (z.B. mit Fuzzing oder State Learning) • SSH (Secure Shell) • Signal • Kryptographische Währungen						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können • Konzepte hinter wichtigsten kryptographischen Protokollen verstehen • Gängige Angriffe auf kryptographische Protokolle verstehen und verhindern • Analysen von kryptographischen Implementierungen mit systematischen Methoden und gängigen Werkzeugen durchführen • Implementierungsfehler und sicherheitstechnische Probleme in kryptographischen Protokollen erkennen und bewerten			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch			
12	Modulbeauftragte/r: Prof. Dr.-Ing. Juraj Somorovsky			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Real World Crypto Engineering:</i> Methodische Umsetzung: Die Inhalte werden durch eine Präsentation in Form einer Vorlesung mit Beamer vermittelt. Dazu werden sie durch individuelle praktische Aufgaben vertieft. Lernmaterialien, Literaturangaben: • Vorlesungsfolien und Übungsblätter • Wissenschaftliche Literatur • Weitere Literatur wird in der Veranstaltung bekanntgegeben.
----	--

3 Wahlpflichtmodule

Reconfigurable Computing							
Reconfigurable Computing							
Modulnummer: M.079.4043		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05703 Reconfigurable Computing	V2 Ü3	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Reconfigurable Computing:</i> Empfohlene Vorkenntnisse Kenntnisse aus den Bachelor-Lehrveranstaltungen Digitaltechnik, Programmierung und Datenstrukturen und Algorithmen sind hilfreich.						

3 Wahlpflichtmodule

4	Inhalte: <i>Inhalte der Lehrveranstaltung Reconfigurable Computing:</i> Die Veranstaltung Reconfigurable Computing führt in das Gebiet des Rechnens mit reprogrammierbaren Hardwarestrukturen ein. Rechnersysteme, die aus reprogrammierbaren Hardwarestrukturen aufgebaut sind, sind nicht auf eine feste Hardware angewiesen, sondern passen ihre Hardwarearchitektur an die auszuführende Anwendung an. Der Bereich entstand Anfang der 1990er Jahre, als Field-programmable Gate Arrays (FPGAs) auf den Markt kamen, die leistungsfähig genug waren, um für das Rechnen verwendet zu werden. Heute übertreffen FPGA-basierte Hochleistungssysteme bei vielen Problemen wie der Datenbanksuche, dem Scannen genomischer Sequenzen und der Kryptografie die modernsten Computer. In eingebetteten Systemen beschleunigen FPGAs die Systemfunktionen, reduzieren die Systemkosten und den Energieverbrauch und ermöglichen Hardware-on-demand-Funktionalität. Die Veranstaltung deckt die folgenden Themen ab: • Einführung in Reconfigurable Computing • Entwicklung von programmierbaren Hardware-Bausteinen • FPGA-Architekturen • Computergestütztes Design für FPGAs • Hochsprachen für die Programmierung von FPGAs • Anwendungsbereiche für FPGAs • Vergleich von reprogrammierbaren Hardwarebausteinen, Technologien und rekonfigurierbaren Systemen								
5	Lernergebnisse und Kompetenzen: Die Studierenden können • verschiedene reprogrammierbare Hardwarebausteine vergleichen und deren geschichtliche Entwicklung schildern, • die Entwurfsschritte und Problemstellungen beim Entwurf mit FPGAs benennen, • Algorithmen für die Entwurfsschritte analysieren und an Beispielen anwenden, • aktuelle Ansätze zur Programmierung von FPGAs vergleichen und bewerten, • die Eignung verschiedener reprogrammierbare Hardwarebausteine für verschiedene Einsatzgebiete begründen und • Funktionen mittlerer Komplexität mit modernen FPGA Entwurfswerkzeugen praktisch umsetzen.								
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>120-180 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote						
a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%						

3 Wahlpflichtmodule

7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch		
12	Modulbeauftragte/r: Prof. Dr. Marco Platzner		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Reconfigurable Computing:</i> Methodische Umsetzung Die Lehrveranstaltung besteht aus einer Vorlesung und Rechen- sowie praktischen Übungen. Die Vorlesung erfolgt mit Beamer und Tafelanschrieb. In den Rechenübungen werden Aufgaben ausgegeben und deren Lösungen werden in einer Übungseinheit vorgestellt und diskutiert. Zusätzlich werden Quizze zur Selbsteinschätzung angeboten. In den praktischen Übungen wird ein Tutorial zum Entwurf mit FPGAs durchgeführt und dann Aufgaben ausgegeben, die in Gruppen von einem bis drei Teilnehmern als Entwurfs- bzw. Programmierbeispiele umgesetzt werden. Lernmaterialien, Literaturangaben • Vorlesungsfolien, Aufgabenblätter für Rechenübungen, Quizze • Tutorial, Aufgabenblätter für Entwurfs- und Programmierbeispiele, technische Dokumentationen • Ausgewählte wissenschaftliche Artikel • Weitere Literatur wird in der Veranstaltung bekanntgegeben.		

3 Wahlpflichtmodule

Resource-Efficient AI: Architectures and Methodologies							
Resource-Efficient AI: Architectures and Methodologies							
Modulnummer: M.048.55106		Workload (h): 180	Leistungspunkte: 6		Turnus: Wintersemester		
		Studiensemester: 1.-3. Semester	Dauer (in Sem.): 1		Sprache: en		
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.048.23023 Resource-Efficient AI: Architectures and Methodologies	2V 2Ü, WS	60	120	P	30/30
2	Wahlmöglichkeiten innerhalb des Moduls: Keine						
3	Teilnahmevoraussetzungen: Keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Resource-Efficient AI: Architectures and Methodologies:</i> Keine						

3 Wahlpflichtmodule

4	Inhalte: <i>Inhalte der Lehrveranstaltung Resource-Efficient AI: Architectures and Methodologies:</i> Die Lehrveranstaltung vermittelt Methoden und Architekturen für die ressourceneffiziente Ausführung moderner KI-Anwendungen unter Berücksichtigung von Leistungs-, Energie- und Speicherrestriktionen. Behandelte Themen umfassen: • Grundlagen des Deep Learning und Trainingsverfahren • Charakteristika moderner KI-Workloads (z. B. CNNs und Transformer) • Rechen-, Speicher- und Energieengpässe bei der KI-Inferenz • Methoden zur effizienten Modellgestaltung und -optimierung: – Quantisierung – Pruning und Sparsity – Knowledge Distillation – Effiziente neuronale Netze (z. B. MobileNet) • Abbildung von KI-Workloads auf unterschiedliche Hardwareplattformen • Architekturen für KI-Beschleunigung: – SIMD- und Vektorarchitekturen – GPU-Architekturen – Dataflow-Architekturen und systolische Arrays – NPUs und domänenspezifische Beschleuniger • Hardware-Software-Co-Design für effiziente KI-Systeme								
5	Lernergebnisse und Kompetenzen: Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage: • grundlegende Prinzipien des Deep Learning aus einer Systemperspektive einzuordnen, • KI-Workloads hinsichtlich ihrer Rechen-, Speicher- und Kommunikationsanforderungen zu analysieren, • Performance-, Energie- und Speicherengpässe zu identifizieren, • Methoden zur Modelloptimierung gezielt anzuwenden, • KI-Workloads auf unterschiedliche Hardwareplattformen abzubilden und zu bewerten, • verschiedene Architekturen für KI-Beschleunigung zu vergleichen, • Trade-offs zwischen Genauigkeit, Latenz, Durchsatz und Energieeffizienz zu analysieren, • Hardware-Software-Co-Design-Methoden für ressourceneffiziente KI-Systeme anzuwenden.								
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur oder mündliche Prüfung oder Referat</td><td>120-180 min oder 30-45 min oder 30 min</td><td>100%</td></tr></table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur oder mündliche Prüfung oder Referat	120-180 min oder 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote						
a)	Klausur oder mündliche Prüfung oder Referat	120-180 min oder 30-45 min oder 30 min	100%						

3 Wahlpflichtmodule

7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Referat	10-20 min
8	Voraussetzungen für die Teilnahme an Prüfungen: Voraussetzung für die Teilnahme an der Modulabschlussprüfung ist das Bestehen der Studienleistung über die Lehrveranstaltung a).		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe der Leistungspunkte erfolgt, wenn die Modulabschlussprüfung (MAP) bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Credits gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Informatik v4		
12	Modulbeauftragte/r: Prof. Dr.-Ing. Lennart Bamberg		
13	Sonstige Hinweise: Modulseite PANDA (mehr Informationen werden in der ersten Vorlesung gegeben) Methodische Umsetzung Vorlesung mit Übung (teilweise mit Simulationen am Rechner) Lernmaterialien, Literaturangaben Handouts und Übungsaufgaben; Literaturhinweise werden in der ersten Vorlesung gegeben		

3 Wahlpflichtmodule

Software Architecture Design and Recovery																					
Software Architecture Design and Recovery																					
Modulnummer: M.079.4094		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester															
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en															
1	Modulstruktur <table><tr><td></td><td>Lehrveranstaltung</td><td>Lehrform</td><td>Kontaktzeit (h)</td><td>Selbststudium (h)</td><td>Status (P/WP)</td><td>Gruppengröße (TN)</td></tr><tr><td>a)</td><td>L.079.05708 Software Architecture Design and Recovery</td><td>V2 Ü3</td><td>75</td><td>105</td><td>P</td><td>30</td></tr></table>								Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)	a)	L.079.05708 Software Architecture Design and Recovery	V2 Ü3	75	105	P	30
	Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)															
a)	L.079.05708 Software Architecture Design and Recovery	V2 Ü3	75	105	P	30															
2	Wahlmöglichkeiten innerhalb des Moduls: keine																				
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Software Architecture Design and Recovery:</i> Empfohlene Vorkenntnisse Ein gutes Verständnis von Java und den Prinzipien objektorientierter Programmierung ist hilfreich.																				
4	Inhalte: <i>Inhalte der Lehrveranstaltung Software Architecture Design and Recovery:</i> Softwarearchitektur beschäftigt sich mit den wichtigsten Entwurfsentscheidungen, die großen Einfluss auf die Qualitätsmerkmale eines Softwaresystems wie Wartbarkeit, Leistung und Sicherheit haben. In dieser Lehrveranstaltung diskutieren wir die Grundlagen der Disziplin Softwarearchitektur sowie Methoden zur Wiederherstellung einer Softwarearchitektur aus existierenden Software-Repositorien. Die Lehrveranstaltung umfasst folgende Inhalte: • Arten von Entwurfsentscheidungen • Architekturkomponenten und Wiederherstellung • Architekturlösungen beispielsweise Muster, Taktiken und Technologien • Architekturdokumentation • Software-Repositorien • Architekturwissen • Architekturprozesse Außerdem werden die folgenden Forschungsmethoden in der Lehrveranstaltung diskutiert und verwendet: • Grounded-theory. • Fallstudien.																				

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden können • Konzepte der Softwarearchitekturdisziplin erklären und diskutieren • große Softwaresysteme für Entwurfsentscheidungen analysieren • Architekturprozesse zum Treffen von Entwurfsentscheidungen ausführen • übliche Forschungsmethode auf Probleme der Softwarearchitektur umsetzen • Forschungsergebnisse zusammenfassen und berichten • in Gruppen arbeiten • einen fachlichen Vortrag halten			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch			
12	Modulbeauftragte/r: Dr. Mohamed Aboubakr Mohamed Soliman			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Software Architecture Design and Recovery:</i> Methodische Umsetzung Vorlesung und Gruppenarbeit mit großen Open-Source-Softwaresysteme sowie Präsentationen. In den Vorlesungen werden Konzepte der Disziplin Softwarearchitektur diskutiert und in einer Reihe von Gruppenarbeiten auf echte Open-Source-Softwaresysteme angewandt. Lernmaterialien, Literaturangaben • Bass, L., Clements, P., Kazman, R. (2012). Software Architecture in Practice. 3rd Edition, Addison-Wesley Professional. • Kruchten P, Lago P, van Vliet H (2006) Building Up and Reasoning About Architectural Knowledge. In: Quality of Software Architectures, Springer Berlin Heidelberg. • Weitere Literatur wird in der Veranstaltung bekanntgegeben.
----	---

3 Wahlpflichtmodule

Software Quality Assurance							
Software Quality Assurance							
Modulnummer: M.079.4048		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester:		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05805 Software Quality Assurance	V3 Ü2	75	105	WP	90/30
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Software Quality Assurance:</i> Empfohlene Vorkenntnisse Programmierung, Modellierung, Modellbasierte Softwareentwicklung						

3 Wahlpflichtmodule

4	Inhalte: <i>Inhalte der Lehrveranstaltung Software Quality Assurance:</i> Das Ziel der Vorlesung ist die Behandlung von Ansätzen, Technologien und Strategien für die Qualitätssicherung von Softwaresystemen. Dies beinhaltet einerseits konstruktive Ansätze wie Design Pattern, Anti-Pattern, domänenspezifische Sprachen, modellgetriebene Softwareentwicklung, Qualitätsmodelle und Architekturstile und andererseits analytische Ansätze wie statische Reviewtechniken und dynamisch Testtechniken. Des Weiteren werden Ansätze für die Verbesserung des Softwareentwicklungsprozesses und internationale Standards wie ISO 9001, 9126, CMM, usw. behandelt. • Einführung in die Softwarequalitätssicherung • Standards – Produktspezifische Standards: ISO 9126 – Prozessspezifische Standards: ISO 9001, CMM • Konstruktive Ansätze – Pattern und Stile: Design Pattern, Anti-Pattern, Architekturstile – Modellgetriebene Softwareentwicklung – Metamodellierung – Domänenspezifische Sprachen – Design By Contract – Forschung: Process Constraints • Analytische Ansätze – Reviews, Inspektionen – Testen: Fundamentaler Testprozess, Black Box Testen, White Box Testen								
5	Lernergebnisse und Kompetenzen: Die Studierenden sind in der Lage, Qualitätseigenschaften von Entwicklungsprozessen, Softwariemodellen bzw. -systemen zu benennen. Sie kennen Techniken zur konstruktiven bzw. analytischen Sicherstellung von Qualitätseigenschaften und können diese geeignet einsetzen. Sie kennen die wesentlichen Standards für die Bewertung von Prozess- und Produktqualitäten. Sie können ausgewählte, aktuelle Forschungsansätze im Bereich Prozess- und Softwarequalitätssicherung verstehen und einordnen.								
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote						
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%						

3 Wahlpflichtmodule

7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Credits gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Informatik v3		
12	Modulbeauftragte/r: Dr. Enes Yigitbas		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Software Quality Assurance:</i> Methodische Umsetzung Eine Mischung aus Folien und Tafelanschrieb. Alle wichtigen Konzepte und Techniken werden in Übungen anhand von Beispielen weiter vertieft. In praktischen Übungen insbesondere mit Testwerkzeugen werden die erlernten Kenntnisse angewendet. Lernmaterialien, Literaturangaben • Daniel Galin: Software Quality Assurance: From Theory to Implementation, Pearson / Addison Wesley, 2004 • Vorlesungsfolien, Übungsaufgaben		

3 Wahlpflichtmodule

Topics in Signal Processing							
Topics in Signal Processing							
Modulnummer: M.048.92014		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1.-3. Semester		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehr- form	Kontakt- zeit (h)	Selbst- studium (h)	Status (P/WP)	Gruppen- größe (TN)
	a)	L.048.92014 Topics in Signal Processing	2V 2Ü, WS	60	120	P	30/30
2	Wahlmöglichkeiten innerhalb des Moduls: Keine						
3	Teilnahmevoraussetzungen: Keine Teilnahmevoraussetzungen der Lehrveranstaltung Topics in Signal Processing: Empfohlen: Signal- und Systemtheorie, mindestens Grundkenntnisse der Wahrscheinlichkeitstheorie und linearen Algebra						
4	Inhalte: Inhalte der Lehrveranstaltung Topics in Signal Processing: Kurzbeschreibung Diese Veranstaltung behandelt eine Auswahl von aktuellen Themen in der Signalverarbeitung. Ein Teil der Veranstaltung besteht aus regulären Vorlesungen, wohingegen der andere aktive Mitarbeit von Studenten voraussetzt. Inhalt Zunächst werden in diesem Kurs relevante Aspekte aus der linearen Algebra und Wahrscheinlichkeitstheorie wiederholt. Danach werden Studenten angeleitet, aktuelle Veröffentlichungen aus der Signalverarbeitungsliteratur zu lesen, zu analysieren und dann auch zu präsentieren.						
5	Lernergebnisse und Kompetenzen: In dieser Veranstaltung werden Studenten mit aktuellen Forschungsthemen in der Signalverarbeitung vertraut gemacht. Studenten lernen, wissenschaftliche Veröffentlichungen zu verstehen und kritisch zu bewerten. Studenten werden das Vertrauen entwickeln, mathematische Probleme in Analyse und Design lösen zu können. Die in dieser Veranstaltung gelernten Prinzipien können auf andere Gebiete angewandt werden.						

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)										
	<table> <tr> <th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr> <tr> <td>a)</td><td>Klausur oder mündliche Prüfung oder Referat</td><td>120-180 min oder 30-45 min oder 30 min</td><td>100%</td></tr> </table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur oder mündliche Prüfung oder Referat	120-180 min oder 30-45 min oder 30 min	100%		
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote								
a)	Klausur oder mündliche Prüfung oder Referat	120-180 min oder 30-45 min oder 30 min	100%								
7	Studienleistung, qualifizierte Teilnahme: keine										
8	Voraussetzungen für die Teilnahme an Prüfungen: Keine										
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe der Leistungspunkte erfolgt, wenn die Modulabschlussprüfung (MAP) bestanden ist.										
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Credits gewichtet (Faktor 1).										
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v3 (CEMA v3), Masterstudiengang Computer Engineering v3 (CEMA v3), englisch, Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Masterstudiengang Informatik v3, Master's Program Electrical Systems Engineering (ESEMA v2), Master's Program Electrical Systems Engineering v3 (ESEMA v3)										
12	Modulbeauftragte/r: Prof. Dr. Peter Schreier										
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Topics in Signal Processing:</i> Lehrveranstaltungsseite http://sst.uni-paderborn.de/teaching/courses/ Methodische Umsetzung Vorlesung und Übung mit aktiver Beteiligung der Studenten, Präsentationen von Studenten Lernmaterialien, Literaturangaben Literaturhinweise werden in der ersten Vorlesung gegeben.										

3 Wahlpflichtmodule

Unsupervised Learning and Evolutionary Optimisation Using R							
Unsupervised Learning and Evolutionary Optimisation Using R							
Modulnummer: M.079.4093		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05829 Unsupervised Learning and Evolutionary Optimisation Using R	V3 Ü2	75	105	P	70
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Unsupervised Learning and Evolutionary Optimisation Using R:</i> Empfohlene Vorkenntnisse • Grundlegende Kenntnisse und Interesse an Mathematik, Statistik und Wahrscheinlichkeitsrechnung • Basiswissen Programmierung						
4	Inhalte: <i>Inhalte der Lehrveranstaltung Unsupervised Learning and Evolutionary Optimisation Using R:</i> Die Veranstaltung vermittelt die formalen und praktischen Grundlagen des unüberwachten maschinellen Lernens sowie deren Umsetzung in der statistischen Programmiersprache R. Insbesondere werden die Methodik und Anwendung der folgenden Themen behandelt: • Einführung in die statistische Programmiersprache R • Datenvorverarbeitung und Qualitätsprüfung von Daten • (Stream-)Clustering-Verfahren • Techniken der Dimensionsreduktion • Grundprinzipien der evolutionären Optimierung, sowohl ein- als auch mehrkriteriell • Praktische Anwendung der Methoden unter Verwendung von R in Einzel- und Gruppenarbeit						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Nach Abschluss des Moduls können die Studierenden ... • die Qualität von Daten erkennen sowie geeignete Techniken zur Datenvorverarbeitung auswählen • wichtige Methoden des unüberwachten Lernens erklären und anwenden • die Grundprinzipien evolutionärer Optimierungsverfahren verstehen • Techniken zur Beurteilung der Qualität von Optimierungsverfahren kompetent anwenden • die Statistik-Software R zur statistischen Datenanalyse, unüberwachtem Lernen und der evolutionären Optimierung anwenden • Problemstellungen im Team analysieren und praxisrelevante Lösungen präsentieren											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><td>zu</td><td>Prüfungsform</td><td>Dauer bzw. Umfang</td><td>Gewichtung für die Modulnote</td></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>90-120 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: <table><tr><td>zu</td><td>Form</td><td>Dauer bzw. Umfang</td><td>SL / QT</td></tr><tr><td>a)</td><td>Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte</td><td></td><td>SL</td></tr></table>				zu	Form	Dauer bzw. Umfang	SL / QT	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL
zu	Form	Dauer bzw. Umfang	SL / QT									
a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte		SL									
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch											
12	Modulbeauftragte/r: Prof. Dr. Heike Trautmann											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Unsupervised Learning and Evolutionary Optimisation Using R:</i> Methodische Umsetzung Eine Einführung in die statistische Programmiersprache R wird kompakt in den ersten Wochen der Veranstaltung durchgeführt. Methoden des unüberwachten maschinellen Lernens werden durch inhaltliche Präsentation und interaktive Übung im Rahmen der Vorlesung vermittelt sowie in Tutorien vertieft, sowohl methodisch formal als auch anwendungsorientiert unter Verwendung von R. Lernmaterialien, Literaturangaben Für die statistische Programmiersprache R wird empfohlen: • Hadley Wickham & Garrett Grolemund (2023). R for Data Science: Import, Tidy, Transform, Visualize, and Model Data. 2nd ed. O'Reilly • Torsten Hothorn and Brian S. Everitt (2014). A Handbook of Statistical Analyses Using R. Chapman & Hall/CRC Press, 3rd edition, 2014. • C. Heumann, M. Schomaker, and Shalabh. Introduction to Statistics and Data Analysis With Exercises, Solutions and Applications in R. Springer, 2017. Die methodischen Kapitel basieren auf der Kombination einer Vielzahl von Literaturquellen, diese werden in der Vorlesung bekannt gegeben.
----	---

3 Wahlpflichtmodule

Usable Security and Privacy								
Usable Security and Privacy								
Modulnummer: M.079.4086		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester		
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en		
1	Modulstruktur							
		Lehrveranstaltung		Lehr- form	Kontakt- zeit (h)	Selbst- studium (h)	Status (P/WP)	Gruppen- größe (TN)
	a)	L.079.05804 Usable Security and Privacy		V2 Ü3	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine							
3	Teilnahmevoraussetzungen: keine							

4	Inhalte: <i>Inhalte der Lehrveranstaltung Usable Security and Privacy:</i> Menschliche Faktoren und Aspekte über die Benutzerfreundlichkeit nehmen in der IT-Sicherheit und in der Entwicklung sicherer Systeme traditionell eine geringe Rolle ein. Thematiken der Benutzerfreundlichkeit werden von Sicherheitsexperten weitgehend vernachlässigt, weil ihre Bedeutung nicht ausreichend geschätzt wird und diese nicht über ausreichende Kenntnisse verfügen, um sie anzugehen. Heute besteht ein Konsens darüber, wie wichtig es ist, das Verhalten der Benutzer zu verstehen und die Benutzerfreundlichkeit zu verbessern, um robuste IT-Sicherheit zu erreichen. Dieser Kurs vermittelt praktisches und forschungsorientiertes Wissen über nutzbare Sicherheit und Privatsphäre. Die Studierenden sammeln praktische Erfahrungen durch Präsenzübungen und arbeiten in kleinen Teams an einem semesterweiten Forschungsprojekt mit dem Ziel, eine Nutzerstudie zu menschenzentrierter Sicherheit und Privatsphäre zu entwerfen und durchzuführen. Dazu werden im Kurs Forschungsmethoden vorgestellt und eine Einführung in HCI- und Usability-Konzepte gegeben. Der Kurs wird sich auch mit grundlegenden und aktuellen Forschungsthemen in diesem Bereich befassen, wie z.B. Tools zur Verbesserung der Privatsphäre und Transparenz, benutzerfreundliche Authentifizierung und entwicklerzentrierte IT-Sicherheit. Durch die Sichtung relevanter Paper und das Halten von Referaten werden die Studierenden mit dem neuesten Stand der Forschung auf diesem Gebiet vertraut gemacht und lernen, wie man wissenschaftlich arbeitet. Die Veranstaltung umfasst folgende Inhalte: • Konzepte für Sicherheit und Datenschutz/Privatsphäre • Grundlagen der Kryptographie • Technologien zur Verbesserung der Privatsphäre und der Transparenz • HCI- und Usability-Forschungsmethoden • Ethik in der Technik • Quantitative und qualitative Datenanalyse • Nutzbare Authentifizierung • Nutzbare Privatsphäre • Entwickler-zentrierte Sicherheit
5	Lernergebnisse und Kompetenzen: Die Studierenden werden • ein Verständnis für die Bedeutung der Sicherheit und des Schutzes der Privatsphäre bei der Nutzung entwickeln • sich mit der Geschichte des Fachgebiets und den wichtigsten Forschungsbereichen und Herausforderungen vertraut machen • in der Lage sein, Methoden zur Durchführung von Nutzerforschung im Bereich Sicherheit und Datenschutz anzuwenden • den aktuellen Forschungsstand zum Thema kennenlernen

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL / QT
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch		
12	Modulbeauftragte/r: Prof. Dr. Patricia Arias Cabarcos		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Usable Security and Privacy:</i> Methodische Umsetzung Grundlegende Konzepte werden in Form einer Vorlesung präsentiert. Durch die Teilnahme an Präsenzübungen und die Durchführung eines Forschungsprojekts in kleinen Gruppen, das sich auf eine Nutzerstudie für die Forschung im Bereich Sicherheit und Privatsphäre konzentriert, können die Studierenden im Laufe des Semesters vertiefte theoretische und praktische Kenntnisse erwerben. Lernmaterialien, Literaturangaben • Lazar, J., Feng, J.H. and Hochheiser, H., 2017. Research methods in human-computer interaction. Morgan Kaufmann. • Redmiles, E.M., Acar, Y., Fahl, S. and Mazurek, M.L., 2017. A summary of survey methodology best practices for security and privacy researchers. • Folien und Verweise auf wissenschaftliche Literatur werden während der Veranstaltung bekanntgegeben.		

3 Wahlpflichtmodule

VLSI-Testing							
VLSI-Testing							
Modulnummer: M.048.92027		Workload (h): 180		Leistungspunkte: 6		Turnus: Wintersemester	
		Studiensemester: 1.-3. Semester		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.048.92027 VLSI Testing	2V 2Ü, WS	60	120	P	30/30
2	Wahlmöglichkeiten innerhalb des Moduls: Keine						
3	Teilnahmevoraussetzungen: Keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung VLSI Testing:</i> Empfohlen: Digitaltechnik						
4	Inhalte: <i>Inhalte der Lehrveranstaltung VLSI Testing:</i> Kurzbeschreibung Die Lehrveranstaltung “VLSI Testing” behandelt systematische Verfahren zur Erkennung von Hardware-Defekten in mikroelektronischen Schaltungen. Es werden sowohl Algorithmen zur Erzeugung und Auswertung von Testdaten als auch Hardwarestrukturen zur Verbesserung der Testbarkeit und für den eingebauten Selbsttest vorgestellt. Inhalt Im einzelnen werden die folgenden Themen behandelt: • Fehlermodelle • Testbarkeitsmaße und Maßnahmen zur Verbesserung der Testbarkeit • Logik- und Fehlersimulation • Algorithmen zur Testmustererzeugung • Selbsttest, insbesondere Testdatenkompression und Testantwortkompaktierung • Speichertest						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Fachkompetenz: Die Studierenden sind nach dem Besuch der Lehrveranstaltung in der Lage, • Fehlermodelle, Maßnahmen zur Verbesserung der Testbarkeit und Werkzeuge zur Unterstützung des Tests zu beschreiben, • die grundlegenden Modelle und Algorithmen für Fehlersimulation und Test zu erklären und anzuwenden, sowie • Systeme im Hinblick auf ihre Testbarkeit zu analysieren und geeignete Teststrategien auszuwählen. Fachübergreifende Kompetenzen: Die Studierenden können • die trainierten Problemlösungsstrategien disziplinübergreifend einsetzen, • ihre Lösungen den anderen Teilnehmern präsentieren und • die erworbenen Kompetenzen im Selbststudium vertiefen.											
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><td>zu</td><td>Prüfungsform</td><td>Dauer bzw. Umfang</td><td>Gewichtung für die Modulnote</td></tr><tr><td>a)</td><td>Klausur oder mündliche Prüfung oder Referat</td><td>120-180 min oder 30-45 min oder 30 min</td><td>100%</td></tr></table>				zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur oder mündliche Prüfung oder Referat	120-180 min oder 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote									
a)	Klausur oder mündliche Prüfung oder Referat	120-180 min oder 30-45 min oder 30 min	100%									
7	Studienleistung, qualifizierte Teilnahme: keine											
8	Voraussetzungen für die Teilnahme an Prüfungen: Keine											
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe der Leistungspunkte erfolgt, wenn die Modulabschlussprüfung (MAP) bestanden ist.											
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Credits gewichtet (Faktor 1).											
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v3 (CEMA v3), Masterstudiengang Computer Engineering v3 (CEMA v3), englisch, Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch, Master's Program Electrical Systems Engineering (ESEMA v2), Master's Program Electrical Systems Engineering v3 (ESEMA v3)											
12	Modulbeauftragte/r: Prof. Dr. Sybille Hellebrand											

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung VLSI Testing:</i> Lehrveranstaltungsseite https://ei.uni-paderborn.de/date/lehre/uebersicht Methodische Umsetzung * Vorlesung mit Beamer und Tafel * Präsenzübungen in kleinen Gruppen mit Übungsblättern zu den theoretischen Grundlagen, Präsentation der Lösungen durch Übungsteilnehmer * Praktische Übungen mit verschiedenen Software-Werkzeugen am Rechner Lernmaterialien, Literaturangaben Aktuelle Hinweise auf ergänzende Literatur und Lehrmaterialien im jeweiligen panda-Kurs • Michael L. Bushnell, Vishwani D. Agrawal, „Essentials of Electronic Testing for Digital, Memory, and Mixed-Signal VLSI Circuits,“ Boston, Dordrecht, London: Kluwer Academic Publishers, 2000 • Laung-Terng Wang, Cheng-Wen Wu, Xiaoqing Wen, „VLSI Test Principles and Architectures: Design for Testability,“ Morgan Kaufmann Series in Systems on Silicon, ISBN: 0123705975
----	--

3 Wahlpflichtmodule

Web Security							
Web Security							
Modulnummer: M.079.4073		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05809 Web Security	V3 Ü2	75	105	P	70/35
2	Wahlmöglichkeiten innerhalb des Moduls: keine						
3	Teilnahmevoraussetzungen: keine <i>Teilnahmevoraussetzungen der Lehrveranstaltung Web Security:</i> Empfohlene Vorkenntnisse Kenntnisse in Programmierung, IT-Sicherheit und Basiskenntnisse in Kryptographie						

4	Inhalte: <i>Inhalte der Lehrveranstaltung Web Security:</i> Moderne Webapplikationen und Webservices sind oft vielschichtig und basieren auf unterschiedlichen (oft komplexen) Technologien, die ständig weiterentwickelt werden. Deren Komplexität ist oft der Grund für neuartige Angriffe, die im Web-Bereich täglich zu sehen sind. In dieser Vorlesung werden wir auf die wichtigsten Technologien eingehen und lernen, worauf man bei der sicheren Web-Entwicklung achten muss. Dabei werden wir prominente und weit verbreitete Angriffe vorstellen und zeigen, wie man die verhindert. Dazu gehören typische Angriffe aus der OWASP Top 10 Liste wie XSS oder SQL Injection bis hin zu Angriffen auf Webservices und Single Sign-On Standards (wie SAML und OpenID Connect). Basierend auf vielen Fällen werden wir lernen, was beim Design und bei der Implementierung von Webapplikationen wichtig ist. Die Veranstaltung umfasst folgende Inhalte: • Einführung in Web Technologien • Web Angriffe – Cross-Site Scripting (XSS) – Cross-Site Request Forgery (CSRF) – Clickjacking – SQL injection • XML und SAML – Angriffe auf XML Parser – Angriffe auf XML Signatur • JSON und OpenID Connect (OIDC) – Angriffe auf OIDC								
5	Lernergebnisse und Kompetenzen: Die Studierenden können • Sicherheitskonzepte hinter Web-Applikationen verstehen • gängige Angriffe auf Web-Applikationen verstehen und verhindern • praktische Analysen von Web-Applikationen mit gängigen Werkzeugen durchführen • Implementierungsfehler und sicherheitstechnische Probleme in Web-Applikationen erkennen und bewerten								
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP) <table><tr><th>zu</th><th>Prüfungsform</th><th>Dauer bzw. Umfang</th><th>Gewichtung für die Modulnote</th></tr><tr><td>a)</td><td>Klausur, mündliche Prüfung oder Referat</td><td>120-180 min, 30-45 min oder 30 min</td><td>100%</td></tr></table>	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote	a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%
zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote						
a)	Klausur, mündliche Prüfung oder Referat	120-180 min, 30-45 min oder 30 min	100%						

3 Wahlpflichtmodule

7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Übungsaufgaben, schriftliche Ausarbeitung oder Fortschrittsberichte	SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: Masterstudiengang Computer Engineering v4 (CEMA v4), Masterstudiengang Computer Engineering v4 (CEMA v4), englisch		
12	Modulbeauftragte/r: Prof. Dr.-Ing. Juraj Somorovsky		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Web Security:</i> Methodische Umsetzung: Die Inhalte werden durch eine Präsentation in Form einer Vorlesung mit Beamer vermittelt. Dazu werden sie durch individuelle praktische Aufgaben vertieft. Lernmaterialien, Literaturangaben: • Vorlesungsfolien und Übungsblätter • Wissenschaftliche Literatur • Weitere Literatur wird in der Veranstaltung bekanntgegeben.		

3 Wahlpflichtmodule

Ausgewählte Themen im Gebiet Classical and Quantum Algorithm Design								
Selected Topics in Classical and Quantum Algorithm Design								
Modulnummer: M.079.4206		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommer- / Wintersemester		
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en		
1	Modulstruktur							
	Lehrveranstaltung			Lehr- form	Kontakt- zeit (h)	Selbst- studium (h)	Status (P/WP)	Gruppen- größe (TN)
2	Wahlmöglichkeiten innerhalb des Moduls: Innerhalb des Moduls ist genau eine der enthaltenen Lehrveranstaltungen zu wählen.							
3	Teilnahmevoraussetzungen: keine							
4	Inhalte: In diesem Modul werden wechselnde Lehrveranstaltungen, beispielsweise von Gastdozenten angeboten. Die konkreten Inhalte sind den Lehrveranstaltungen zu entnehmen.							
5	Lernergebnisse und Kompetenzen: Die Studierenden sind in der Lage für ein spezifisches Themenfeld aus dem Gebiet Classical and Quantum Algorithm Design: • Problemstellungen adäquat zu beschreiben • Problemstellungen zu analysieren und Lösungsansätze zu erarbeiten • Lösungen zu gegebenen Problemstellungen zu erarbeiten und umzusetzen • Lösungen zu gegebenen Problemstellungen zu bewerten • Verschiedene Lösungen zu gegebenen Problemstellungen zu vergleichen, zu bewerten und eine adäquate Lösung auszuwählen							
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)							
	zu	Prüfungsform			Dauer bzw. Umfang		Gewichtung für die Modulnote	
		Klausur, mündliche Prüfung oder Referat			90-120 min, 30-45 min oder 30 min		100%	
7	Studienleistung, qualifizierte Teilnahme: keine							

3 Wahlpflichtmodule

8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).
11	Verwendung des Moduls in anderen Studiengängen: keine
12	Modulbeauftragte/r: Studiengangsbeauftragter Informatik
13	Sonstige Hinweise: keine

3 Wahlpflichtmodule

Ausgewählte Themen im Gebiet Computer and Communication Systems							
Selected Topics in Computer and Communication Systems							
Modulnummer: M.079.4207		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommer- / Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
	Lehrveranstaltung		Lehr- form	Kontakt- zeit (h)	Selbst- studium (h)	Status (P/WP)	Gruppen- größe (TN)
2	Wahlmöglichkeiten innerhalb des Moduls: Innerhalb des Moduls ist genau eine der enthaltenen Lehrveranstaltungen zu wählen.						
3	Teilnahmevoraussetzungen: keine						
4	Inhalte: In diesem Modul werden wechselnde Lehrveranstaltungen, beispielsweise von Gastdozenten angeboten. Die konkreten Inhalte sind den Lehrveranstaltungen zu entnehmen.						
5	Lernergebnisse und Kompetenzen: Die Studierenden sind in der Lage für ein spezifisches Themenfeld aus dem Gebiet Computer and Communication Systems: • Problemstellungen adäquat zu beschreiben • Problemstellungen zu analysieren und Lösungsansätze zu erarbeiten • Lösungen zu gegebenen Problemstellungen zu erarbeiten und umzusetzen • Lösungen zu gegebenen Problemstellungen zu bewerten • Verschiedene Lösungen zu gegebenen Problemstellungen zu vergleichen, zu bewerten und eine adäquate Lösung auszuwählen						
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)						
	zu	Prüfungsform			Dauer bzw. Umfang		Gewichtung für die Modulnote
		Klausur, mündliche Prüfung oder Referat			90-120 min, 30-45 min oder 30 min		100%
7	Studienleistung, qualifizierte Teilnahme: keine						

3 Wahlpflichtmodule

8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).
11	Verwendung des Moduls in anderen Studiengängen: keine
12	Modulbeauftragte/r: Studiengangsbeauftragter Informatik
13	Sonstige Hinweise: keine

3 Wahlpflichtmodule

Ausgewählte Themen im Gebiet Data Science and Intelligent Systems							
Selected Topics in Data Science and Intelligent Systems							
Modulnummer: M.079.4208		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommer- / Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
	Lehrveranstaltung		Lehr- form	Kontakt- zeit (h)	Selbst- studium (h)	Status (P/WP)	Gruppen- größe (TN)
2	Wahlmöglichkeiten innerhalb des Moduls: Innerhalb des Moduls ist genau eine der enthaltenen Lehrveranstaltungen zu wählen.						
3	Teilnahmevoraussetzungen: keine						
4	Inhalte: In diesem Modul werden wechselnde Lehrveranstaltungen, beispielsweise von Gastdozenten angeboten. Die konkreten Inhalte sind den Lehrveranstaltungen zu entnehmen.						
5	Lernergebnisse und Kompetenzen: Die Studierenden sind in der Lage für ein spezifisches Themenfeld aus dem Gebiet Data Science and Intelligent Systems: • Problemstellungen adäquat zu beschreiben • Problemstellungen zu analysieren und Lösungsansätze zu erarbeiten • Lösungen zu gegebenen Problemstellungen zu erarbeiten und umzusetzen • Lösungen zu gegebenen Problemstellungen zu bewerten • Verschiedene Lösungen zu gegebenen Problemstellungen zu vergleichen, zu bewerten und eine adäquate Lösung auszuwählen						
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)						
	zu	Prüfungsform			Dauer bzw. Umfang	Gewichtung für die Modulnote	
		Klausur, mündliche Prüfung oder Referat			90-120 min, 30-45 min oder 30 min	100%	
7	Studienleistung, qualifizierte Teilnahme: keine						

3 Wahlpflichtmodule

8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).
11	Verwendung des Moduls in anderen Studiengängen: keine
12	Modulbeauftragte/r: Studiengangsbeauftragter Informatik
13	Sonstige Hinweise: keine

3 Wahlpflichtmodule

Ausgewählte Themen im Gebiet Security							
Selected Topics in Security							
Modulnummer: M.079.4083		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommer- / Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
	Lehrveranstaltung		Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
a)	L.079.05798 Privacy Enhancing Technologies		V2 Ü3	75	105	WP	40
2	Wahlmöglichkeiten innerhalb des Moduls: Innerhalb des Moduls ist genau eine der enthaltenen Lehrveranstaltungen zu wählen.						
3	Teilnahmevoraussetzungen: keine						
4	Inhalte: In diesem Modul werden wechselnde Lehrveranstaltungen, beispielsweise von Gastdozenten angeboten. Die konkreten Inhalte sind den Lehrveranstaltungen zu entnehmen. <i>Inhalte der Lehrveranstaltung Privacy Enhancing Technologies:</i> Dieser Kurs vermittelt den Teilnehmern ein grundlegendes Verständnis der Risiken für die Privatsphäre, der gängigsten Technologien zu deren Bewältigung ("PETS") und der menschlichen Faktoren, die deren Gestaltung beeinflussen.						
5	Lernergebnisse und Kompetenzen: Die Studierenden sind in der Lage für ein spezifisches Themenfeld aus dem Gebiet Security: • Problemstellungen adäquat zu beschreiben • Problemstellungen zu analysieren und Lösungsansätze zu erarbeiten • Lösungen zu gegebenen Problemstellungen zu erarbeiten und umzusetzen • Lösungen zu gegebenen Problemstellungen zu bewerten • Verschiedene Lösungen zu gegebenen Problemstellungen zu vergleichen, zu bewerten und eine adäquate Lösung auszuwählen						

3 Wahlpflichtmodule

6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)		
	zu	Prüfungsform	Dauer bzw. Umfang
	a)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min
7	Studienleistung, qualifizierte Teilnahme:		
	zu	Form	Dauer bzw. Umfang
	a)	Praktische Arbeiten und Diskussion	SL / QT
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung		
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.		
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).		
11	Verwendung des Moduls in anderen Studiengängen: keine		
12	Modulbeauftragte/r: Studiengangsbeauftragter Informatik		
13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Privacy Enhancing Technologies:</i> Methodische Umsetzung • Wöchentliche Theorievorlesung • Übung und Aufgaben zur interaktiven Diskussion in den praktischen Vorlesungen Lernmaterialien, Literaturangaben • Vorlesungsfolien, wissenschaftliche Literatur und spezifische Lektüre werden während des Kurses zur Verfügung gestellt.		

3 Wahlpflichtmodule

Ausgewählte Themen im Gebiet Software Engineering							
Selected Topics in Software Engineering							
Modulnummer: M.079.4078		Workload (h): 180		Leistungspunkte: 6		Turnus: Sommer- / Wintersemester	
		Studiensemester: 1-3		Dauer (in Sem.): 1		Sprache: en	
1	Modulstruktur						
		Lehrveranstaltung	Lehrform	Kontaktzeit (h)	Selbststudium (h)	Status (P/WP)	Gruppengröße (TN)
	a)	L.079.05796 Software Engineering for Self-Adaptive Systems	V3 Ü2	75	105	WP	40
	b)	L.079.05798 Privacy Enhancing Technologies	V2 Ü3	75	105	WP	40
2	Wahlmöglichkeiten innerhalb des Moduls: Innerhalb des Moduls ist genau eine der enthaltenen Lehrveranstaltungen zu wählen.						
3	Teilnahmevoraussetzungen: <i>Teilnahmevoraussetzungen der Lehrveranstaltung Software Engineering for Self-Adaptive Systems:</i> Empfohlene Vorkenntnisse Programmieren und Modellieren						
4	Inhalte: In diesem Modul werden wechselnde Lehrveranstaltungen, beispielsweise von Gastdozenten angeboten. Die konkreten Inhalte sind den Lehrveranstaltungen zu entnehmen. <i>Inhalte der Lehrveranstaltung Software Engineering for Self-Adaptive Systems:</i> Die Komplexität von Softwaresystemen, die Evolution von Anforderungen und die Unsicherheit von Anforderungen und Umgebungen erschweren das Management laufender Systeme. Eine vielversprechende Lösung sind selbst-adaptive Systeme, die in der Lage sind, sich automatisch an sich ändernden Anforderungen und Umgebungen anzupassen. In der Vorlesung werden die Grundlagen als auch Methoden und Techniken für die Entwicklung und Absicherung selbst-adaptiver Software behandelt. Der Fokus liegt dabei auf dem Software Engineering für selbst-adaptive Systeme und der Auswirkung der Selbst-Adaption auf die Engineering-Aktivitäten. Auf Basis von Forschungsprototypen (sog. "Exemplars") werden ausgewählte Methoden und Techniken in den Übungen und im Projekt erprobt. <i>Inhalte der Lehrveranstaltung Privacy Enhancing Technologies:</i> Dieser Kurs vermittelt den Teilnehmern ein grundlegendes Verständnis der Risiken für die Privatsphäre, der gängigsten Technologien zu deren Bewältigung ("PETS") und der menschlichen Faktoren, die deren Gestaltung beeinflussen.						

3 Wahlpflichtmodule

5	Lernergebnisse und Kompetenzen: Die Studierenden sind in der Lage für ein spezifisches Themenfeld aus dem Gebiet Software Engineering: • Problemstellungen adäquat zu beschreiben • Problemstellungen zu analysieren und Lösungsansätze zu erarbeiten • Lösungen zu gegebenen Problemstellungen zu erarbeiten und umzusetzen • Lösungen zu gegebenen Problemstellungen zu bewerten • Verschiedene Lösungen zu gegebenen Problemstellungen zu vergleichen, zu bewerten und eine adäquate Lösung auszuwählen			
6	Prüfungsleistung: ☒ Modulabschlussprüfung (MAP) ☐ Modulprüfung (MP) ☐ Modulteilprüfungen (MTP)			
	zu	Prüfungsform	Dauer bzw. Umfang	Gewichtung für die Modulnote
	a) - b)	Klausur, mündliche Prüfung oder Referat	90-120 min, 30-45 min oder 30 min	100%
7	Studienleistung, qualifizierte Teilnahme:			
	zu	Form	Dauer bzw. Umfang	SL / QT
	a)	Schriftliche Übungsaufgaben		SL
	b)	Praktische Arbeiten und Diskussion		SL
8	Voraussetzungen für die Teilnahme an Prüfungen: Bestehen der Studienleistung			
9	Voraussetzungen für die Vergabe von Credits: Die Vergabe von Credits erfolgt, wenn die Modulabschlussprüfung bestanden ist.			
10	Gewichtung für Gesamtnote: Das Modul wird mit der Anzahl seiner Leistungspunkte gewichtet (Faktor 1).			
11	Verwendung des Moduls in anderen Studiengängen: keine			
12	Modulbeauftragte/r: Studiengangsbeauftragter Informatik			

13	Sonstige Hinweise: <i>Hinweise der Lehrveranstaltung Software Engineering for Self-Adaptive Systems:</i> Methodische Umsetzung Die Grundlagen, Methoden und Techniken für die Entwicklung und Absicherung selbst-adaptiver Systeme werden in der Vorlesung vermittelt. In der Übung/Projekt werden ausgewählte Methoden und Techniken an sogenannten Exemplars aus der Forschung erprobt (http://self-adaptive.org/exemplars/). Lernmaterialien, Literaturangaben Folien, Hinweise zur wissenschaftlichen Papieren werden in der Vorlesung gegeben, Bücher: • Weyns, Danny (2020). An Introduction to Self-adaptive Systems: A Contemporary Software Engineering Perspective. Wiley • Cheng, B.H.C., de Lemos, R., Inverardi, P., Magee, J. (Eds.). (2009) Software Engineering for Self-Adaptive Systems. Springer. • de Lemos, R., Giese, H., Müller, H., Shaw, M. (Eds.) (2013). Software Engineering for Self-Adaptive Systems II. Springer. • de Lemos, R., Garlan, D., Ghezzi, C., Giese, H. (Eds.) (2017). Software Engineering for Self-Adaptive Systems III. Assurances. Springer <i>Hinweise der Lehrveranstaltung Privacy Enhancing Technologies:</i> Methodische Umsetzung • Wöchentliche Theorievorlesung • Übung und Aufgaben zur interaktiven Diskussion in den praktischen Vorlesungen Lernmaterialien, Literaturangaben • Vorlesungsfolien, wissenschaftliche Literatur und spezifische Lektüre werden während des Kurses zur Verfügung gestellt.
----	--

4 Focus Areas

4.1 Classical and Quantum Algorithm Design

Koordination

Prof. Dr. Christian Scheideler

Module in dieser Focus Area

- Advanced Algorithms
- Advanced Complexity Theory
- Advanced Distributed Algorithms and Data Structures
- Algorithms for Complex Virtual Scenes
- Foundational Methods for Knowledge Representation and Reasoning
- Foundations of Cryptography
- Geometric Deep Learning
- Geometry Processing
- Introduction to Description Logics
- Introduction to Quantum Computation
- Post-Quantum Cryptography
- Quantum Algorithms
- Quantum Complexity Theory
- Quantum Information

Beschreibung

In dieser Focus Area lernen Studierende wesentliche Techniken zum Entwurf effizienter Algorithmen kennen, sowohl für klassische sequentielle und verteilte Modelle als auch für Quantum Computing. Anwendungsgebiete sind dabei unter anderem effiziente Algorithmen für Computergrafik, verteilte Systeme, Optimierungsprobleme, Spieltheorie und Big Data. Darüber hinaus werden die Grenzen für den Entwurf effizienter Algorithmen im Bereich der klassischen Komplexitätstheorie und des Quantum Computing ausgelotet. Weiterhin werden konstruktive Anwendungen der Grenzen des Entwurfs effizienter Algorithmen im Bereich der klassischen Kryptographie, Post-Quantum Kryptographie und der IT-Sicherheit vorgestellt.

4.2 Computer and Communication Systems

Koordination

Prof. Dr. Marco Platzner

Module in dieser Focus Area

- Advanced Computer Architecture
- Advanced Distributed Algorithms and Data Structures
- Advanced Networked Systems
- Approximate Computing
- Model-Based Systems Engineering
- Real-Time Operating Systems: Designs and Implementations
- Reconfigurable Computing
- Resource-Efficient AI: Architectures and Methodologies
- VLSI Testing
- Web Security

Beschreibung

Computer- und Kommunikationstechnologien sind nicht nur zwei klassische Teilgebiete der Informatik sondern stellen auch die technische Infrastruktur für alle Anwendungen der Informatik bereit. Beide Teilgebiete haben sich in den letzten Jahrzehnten rasant entwickelt und insbesondere die zunehmende Verschmelzung von Computing und Communication ist die Grundlage vieler moderner Informatiksysteme.

Das Vertiefungsgebiet „Computer and Communication Systems“ behandelt vertiefend und im technischen Detail verschiedene Aspekte von modernen Computer- und Kommunikationssystemen. Bei den Computersystemen stehen dabei die Analyse und Bewertung von Rechnerarchitekturen, systematische Methoden für den Entwurf und die Optimierung von Computersystemen, insbesondere das Zusammenspiel von Hardware und Software, sowie Programmiermodelle und -methoden für die stark an Bedeutung gewinnenden parallelen und spezialisierten Rechnerarchitekturen im Vordergrund. Bei den Kommunikationssystemen werden Architekturen, Methoden und Systeme moderner Kommunikationstechnik auf unterschiedlichen Abstraktionsebenen untersucht, beginnend bei der physikalischen Übertragung bis hin zum Anwendungsentwurf in verteilten Umgebungen. Dabei werden unterschiedliche Systemklassen behandelt, von klassischer Mobilkommunikation über ad hoc Netze und Fahrzeugkommunikation bis zur Vernetzung in Rechenzentren und Architekturen des zukünftigen Internets.

4.3 Data Science and Intelligent Systems

Koordination

Prof. Dr. Axel-Cyrille Ngonga Ngomo

Module in dieser Focus Area

- Advanced Algorithms
- Advanced Distributed Algorithms and Data Structures
- Data Science for Software Engineering
- Data Science in Industrial Applications
- Digitale Sprachsignalverarbeitung
- Explainable Artificial Intelligence
- Foundational Methods for Knowledge Representation and Reasoning
- Foundations of Knowledge Graphs
- Introduction to Description Logics
- Machine Learning 1
- Multi-Objective Optimisation
- Topics in Signal Processing
- Unsupervised Learning and Evolutionary Optimisation Using R

Beschreibung

Datenintensive Systeme werden in einer zunehmenden Anzahl von Anwendungsgebieten eingesetzt. In dieser Focus Area werden Algorithmen und Architekturen untersucht, die diesen Systemen zugrunde liegen. Die Focus Area fußt auf zwei nicht disjunkten Forschungsgebieten: Data Science und intelligente Systeme.

Data Science ist eine junge wissenschaftliche Disziplin im Schnittbereich von Informatik, Statistik, Mathematik und den Ingenieurwissenschaften, die sich in den letzten Jahren zu einem der einflussreichsten Gebiete der Forschungslandschaft entwickelt hat. Sie trägt maßgeblichen Anteil an der Digitalisierung und "Datafizierung" unserer Gesellschaft, nicht nur in der Industrie und Forschung sondern auch im privaten Umfeld. In der Wissenschaft wird sie neben dem empirischen, theoretischen und computationalen Ansatz oft als "viertes Paradigma" betrachtet. Grob gesagt besteht das Ziel der Data Science in der Entwicklung methodischer und algorithmischer Grundlagen einer automatisierten Erzeugung nützlichen Wissens aus Daten, sowie der Umsetzung entsprechender Grundlagen in Form von Computersystemen. Das Vertiefungsgebiet "Data Science" stattet die Studierenden mit solidem theoretischem Grundwissen sowie praktischen Fertigkeiten aus, die ihnen das Profil eines modernen "Data Scientist" verleihen. Hierzu werden Veranstaltungen in drei Richtungen angeboten: Mathematische und algorithmische Grundlagen, Data Analytics, Software und Systeme.

Von besonderer Bedeutung für die Focus Area sind intelligente Systeme, ergo Systeme, deren Verhalten durch Methoden und Algorithmen der Künstlichen Intelligenz (KI) gesteuert wird. Solche Systeme gewinnen kontinuierlich an Bedeutung, nicht nur auf wissenschaftlicher Ebene innerhalb der Informatik, sondern auch im sozialen und gesellschaftlichen Kontext: Autonome oder teilautonome Systeme wie Serviceroboter, selbstfahrende PKWs oder medizinische Diagnosesysteme verändern zunehmend unser privates und berufliches Leben. Neben methodischen Fortschritten und einer Steigerung der Rechenleistung durch schnellere Hardware ist die rasante Entwicklung von KI-Systemen in der letzten Dekade vor allem einer Datenexplosion zu verdanken: Die Verfügbarkeit großer Mengen von Daten oder sensorisch erfasster Beobachtungen aus ihrer Umgebung versetzt intelligente Systeme in die La-

ge, ihr Verhalten durch Adaption und Lernen selbständig zu optimieren. Diese Focus Area greift wichtige Aspekte des Entwurfs intelligenter Systeme auf und vermittelt entsprechende theoretische und methodische Grundlagen. Die Inhalte erstrecken sich von Themen wie Maschinelles Lernen bis zum Datenmanagement von strukturierten Daten für erklärbare KI.

4.4 Security

Koordination

Prof. Dr. Yasemin Acar

Module in dieser Focus Area

- Advanced Distributed Algorithms and Data Structures
- Designing code analyses for large-scale software systems 1
- Designing code analyses for large-scale software systems 2
- Foundations of Cryptography
- Hardware Security
- Human Factors in Security and Privacy
- Introduction to Quantum Computation
- Post-Quantum Cryptography
- Privacy and Technology
- Quantum Complexity Theory
- Real World Crypto Engineering
- Usable Security and Privacy
- Web Security

Beschreibung

In allen Lebensbereichen bieten digitale Technologien, wie zum Beispiel das (Industrial) Internet of Things, Cyber-Physical Systems, Digital Automotives, Digital Health oder Industrie 4.0, ein immenses Innovationspotenzial. Die zunehmende Digitalisierung erfordert jedoch neue Ansätze, um dieses Potenzial sicher nutzen zu können. Um diese Herausforderung angehen zu können, besteht in Industrie, Forschung und Lehre ein großer Bedarf an gut ausgebildeten Informatik-Expert*innen mit fundierten Kenntnissen in der IT-Sicherheit. Im Vertiefungsgebiet "IT Security" wird ein solides theoretisches Grundwissen in Kombination mit praktischen Fertigkeiten vermittelt. Das Lehrangebot deckt fachliche Kompetenzen aus dem Bereich der IT-Sicherheit ab (z.B. Softwaresicherheit, formale Verifikation, Grundlagen der modernen Kryptographie und Kommunikationssicherheit) ab, in denen typische Sicherheitslücken und Angriffstechniken vorgestellt werden und Gegenmaßnahmen und ihre Wirksamkeit untersucht werden.

Da Sicherheit nicht unabhängig von konkreten Anwendungen gesehen werden kann und unterschiedliche Anwendungen unterschiedliche Sicherheitsanforderungen haben, werden darüber hinaus auch fachliche Kompetenzen in modernen Anwendungsfeldern mit besonderen Sicherheitsanforderungen

(z.B. Kommunikationsprotokolle in den Bereichen Mobile und Automotive) sowie ergänzende Qualifikationen in den Bereichen Algorithmen und Quanten-Computing abgedeckt.

4.5 Software Engineering

Koordination

Prof. Dr. Eric Bodden

Module in dieser Focus Area

- Agentic Software Engineering
- Data-Driven Engineering
- Data-Driven Innovation
- Data Science for Software Engineering
- Data Science in Industrial Applications
- Designing code analyses for large-scale software systems 1
- Designing code analyses for large-scale software systems 2
- Human Factors in Security and Privacy
- Model-Based Systems Engineering
- Software Architecture Design and Recovery
- Software Quality Assurance

Beschreibung

Software durchdringt schon seit langem kontinuierlich alle Bereiche des privaten, professionellen und öffentlichen Lebens. Ohne effektive, effiziente und zuverlässige Software die unsere Welt, wie wir sie kennen, undenkbar. Aber wie baut man aktuelle Software so, dass sie zuverlässig und effizient auch große Datenmengen verarbeiten kann? Und wie baut man hoch qualitative Software mit aktuellen Methoden und prüft ihre Eigenschaften auf möglichst automatisierte Art und Weise? Und wie begegnet man dem wichtigen Faktor Mensch in der Softwareentwicklung?

Im Modul Software Engineering erlernen Sie eine aktuelle, forschungsnahe Sicht auf diese Fragestellungen. Sie lernen aktuelle Herausforderungen kennen, und mit welchen Methoden und Werkzeugen das Software Engineering diesen begegnet. Hierzu erlernen Sie zum einen konstruktive Methoden zur Realisierung von funktionalen und nicht-funktionalen Aspekten softwareintensiver Systeme, aber auch analytische Methoden, mit welchen Sie die effektive Umsetzung und Zielerreichung in Bezug auf diese Aspekte nachhaltig prüfen können. Sie erlangen weiterhin Tiefenwissen über die verschiedenen Rollen in der Softwareentwicklung und welche Faktoren in der Zusammenarbeit dieser Rollen welche Einflüsse haben können, sowie sie diese mittels empirischer Studien systematisch erfassen können.

5 Module im Wintersemester

• M.079.4005 Advanced Computer Architecture	22
• M.079.4006 Advanced Distributed Algorithms and Data Structures	25
• M.079.4212 Agentic Software Engineering	31
• M.079.4206 Ausgewählte Themen im Gebiet Classical and Quantum Algorithm Design	145
• M.079.4207 Ausgewählte Themen im Gebiet Computer and Communication Systems	147
• M.079.4208 Ausgewählte Themen im Gebiet Data Science and Intelligent Systems	149
• M.079.4083 Ausgewählte Themen im Gebiet Security	151
• M.079.4078 Ausgewählte Themen im Gebiet Software Engineering	153
• M.079.4070 Designing code analyses for large-scale software systems 1	48
• M.079.4054 Foundations of Knowledge Graphs	68
• General Studies	14
• M.079.4209 Geometric Deep Learning	70
• M.079.4092 Human Factors in Security and Privacy	79
• M.079.4059 Introduction to Quantum Computation	85
• M.079.4202 Key Skills	11
• M.079.4032 Machine Learning 1	90
• A.079.4010 Master Thesis	4
• M.079.4062 Model-Based Systems Engineering	93
• M.079.4087 Privacy and Technology	102
• M.079.4201 Project Group	8
• M.079.4090 Quantum Information	110
• M.079.4067 Real World Crypto Engineering	116
• M.079.4213 Real-Time Operating Systems: Design and Implementations	113
• M.079.4043 Reconfigurable Computing	119
• M.048.55106 Resource-Efficient AI: Architectures and Methodologies	122
• M.079.4094 Software Architecture Design and Recovery	125
• M.048.92014 Topics in Signal Processing	131
• M.048.92027 VLSI-Testing	139

6 Module im Sommersemester

• M.079.4002 Advanced Algorithms	16
• M.079.4004 Advanced Complexity Theory	19
• M.079.4096 Advanced Networked Systems	28
• M.079.4009 Algorithms for Complex Virtual Scenes	35
• M.079.4068 Approximate Computing	39
• M.079.4206 Ausgewählte Themen im Gebiet Classical and Quantum Algorithm Design	145
• M.079.4207 Ausgewählte Themen im Gebiet Computer and Communication Systems	147
• M.079.4208 Ausgewählte Themen im Gebiet Data Science and Intelligent Systems	149
• M.079.4083 Ausgewählte Themen im Gebiet Security	151
• M.079.4078 Ausgewählte Themen im Gebiet Software Engineering	153
• M.079.4101 Data Science for Software Engineering	45
• M.079.4076 Data-Driven Innovation	42
• M.079.4071 Designing code analyses for large-scale software systems 2	52
• M.048.24001 Digitale Sprachsignalverarbeitung	56
• M.079.4091 Explainable Artificial Intelligence	59
• M.079.4210 Foundational Methods for Knowledge Representation and Reasoning	62
• M.079.4020 Foundations of Cryptography	65
• General Studies	14
• M.079.4205 Geometry Processing	73
• M.079.4211 Hardware Security	76
• M.079.4098 Introduction to Description Logics	82
• M.079.4202 Key Skills	11
• M.079.4031 Logic Programming for Artificial Intelligence	87
• A.079.4010 Master Thesis	4
• M.079.4095 Multi-Objective Optimisation	96
• M.079.4089 Post-Quantum Cryptography	99
• M.079.4201 Project Group	8
• M.079.4072 Quantum Algorithms	105
• M.079.4063 Quantum Complexity Theory	107
• M.079.4048 Software Quality Assurance	128
• M.079.4093 Unsupervised Learning and Evolutionary Optimisation Using R	133
• M.079.4086 Usable Security and Privacy	136
• M.079.4073 Web Security	142

Erzeugt am 10. Juli 2026 um 15:31.