

Nebenfach Informatik

Das Nebenfach besteht aus dem Pflichtmodul PRG1, das um andere Module aus dem Katalog ergänzt werden kann. Informatik kann sowohl im Bachelor als auch im Master gewählt werden. Im Master dürfen nur Module verwendet werden, die noch nicht im Bachelor eingebracht wurden. Prüfungen und deren Benotung erfolgen nach der Ordnung für den Bachelor und Master Informatik.

Modul PRG1 Programmierung 1 (9 CP)

Inhalt, Ziel: Es sollen die grundlegenden Sprachparadigmen und -konzepte für Algorithmen, Programme und Daten verstanden und gelernt werden. Der Unterschied zwischen Syntax und Semantik einer Programmiersprache sollte verstanden werden. Die Studierenden sollen die Fähigkeit erwerben, die Struktur, das Design und den Einsatzbereich verschiedener Programmiersprachen zu erkennen und einschätzen zu können, und sollen in die Lage versetzt werden, verschiedene, auch zukünftige Programmiersprachen selbstständig zu erlernen, auf ihre Eignung für bestimmte Einsatzgebiete beurteilen sowie Software-Entwürfe auf Programmierkonzepte abbilden zu können. Die Studierenden sollen den Lebenszyklus von Software und elementare Prozesse und Methoden der Software-Entwicklung kennen lernen.

Weiterhin sollen die typischen Konzepte und Eigenschaften von Betriebssystemen kennengelernt werden, um bei Problemen konstruktiv eingreifen zu können. Die Studierenden sollen dabei auch für das Problemfeld der IT-Sicherheit sensibilisiert werden. Die Studierenden sollen Grundkenntnisse von Netzwerken und verteilten Systemen: Dienste und Protokolle, Kommunikationssysteme, Internet, Netzarchitekturen und Netzsicherheit kennen lernen.

Turnus: Jährlich, Beginn in jedem Wintersemester

Voraussetzungen für die Teilnahme: -

Prüfungsvorleistungen: -

Modulprüfung, Prüfungsform: 120-minütige Klausur, benotet

Für die Prüfung verantwortlicher Fachbereich: Informatik

Voraussetzung für die Vergabe der Leistungspunkte des Moduls: Bestehen der Modulabschlussprüfung

Verwendbarkeit: Bachelor. Master, soweit nicht schon im Bachelor eingebracht

					Semester/CP
Lehrveranstaltung	Turnus	Typ	SWS	Jedes WS	
Grundlagen der Programmierung 1	SS	V	4	9	-
Übungen zur Vorlesung		Ü	2		
Lehrveranstaltung:	Grundlagen der Programmierung 1				
Inhalt:	Grundlegende Elemente und Konzepte imperativer und objektorientierter Programmiersprachen: Datenstrukturen, Kontrollstrukturen, Datentypen; vom Problem zum Algorithmus, Algorithmenentwurf. Elemente des Softwareengineerings: Entwicklungszyklen, Modularisierung, Anforderungen, Spezifikation, Korrektheit, Testen, Dokumentation. Grundlagen von Betriebssystemen: Aufgaben und Struktur, Prozesse, Nebenläufigkeit, Synchronisation und Kommunikation, Dateien und Dateisysteme, Sicherheit und Schutzmechanismen, Systemaufrufe. Rechnernetze und Verteilte Systeme: Dienste und Protokolle, Kommunikationssysteme, Internet, Netzarchitekturen und Netzsicherheit.				
Vorkenntnisse:	-1				
Studiennachweis:	Regelmäßige und erfolgreiche Teilnahme an den Übungen				
Prüfung:	Klausur zur Veranstaltung ist gleichzeitig die Modulabschlussprüfung (s.o.)				

Modul PRG2 Programmierung 2 (9 CP)

Inhalt, Ziel: Es sollen die grundlegenden Sprachparadigmen und -konzepte für die Programmierung verstanden und gelernt werden. Die Studierenden sollen die Fähigkeit vertiefen, die Struktur, das Design und den Einsatzbereich verschiedener Programmiersprachen zu erkennen und einschätzen zu können, und sollen in die Lage versetzt werden, verschiedene, auch zukünftige Programmiersprachen selbstständig zu erlernen, auf ihre Eignung für bestimmte Einsatzgebiete beurteilen sowie Software-Entwürfe auf Programmierkonzepte abbilden zu können.

Weiterhin sollen die typischen Konzepte und Eigenschaften von Datenbanken kennen gelernt werden, um bei Problemen konstruktiv eingreifen zu können.

Turnus: Jährlich, Beginn in jedem Sommersemester

Voraussetzungen für die Teilnahme: Modul PRG

Prüfungsvorleistungen: -

Modulprüfung, Prüfungsform: 160-minütige Klausur, benotet

Für die Prüfung verantwortlicher Fachbereich: Informatik

Voraussetzung für die Vergabe der Leistungspunkte des Moduls: Bestehen der Modulabschlussprüfung

Verwendbarkeit: Bachelor, Master, soweit nicht schon im Bachelor eingebracht

					Semester/CP
Lehrveranstaltung	Turnus	Typ	SWS	Jedes WS	Folgendes SS
Grundlagen der Programmierung 2	SS	V	3	9	-
Übungen zur Vorlesung		Ü	2		
Lehrveranstaltung:	Grundlagen der Programmierung 2				
Inhalt:	Übersicht über Sprachparadigmen: Funktionale Programmierung, Rekursion und Iteration, Typisierung, Operationale Semantik für funktionale Programmiersprachen, parallele Programmierkonzepte. Einführung in die objektorientierte Programmierung: Klassen, Objekte, Kommunikation, Vererbung, Architekturen von OO-Programmen. Einführung in Datenbanksysteme: Architekturen, konzeptionelle und logische Modelle, Entity-Relationship-Modell, Relationenmodell, Normalformen, Datenbankdesign, Abfragesprachen (SQL)				
Vorkenntnisse:	PRG1				
Studiennachweis:	Regelmäßige und erfolgreiche Teilnahme an den Übungen				
Prüfung:	Klausur zu Veranstaltung ist gleichzeitig die Modulabschlussprüfung (s.o.)				

Modul HRD Hardware-Architektur und Rechensysteme (8 CP)

Inhalt, Ziel: Modellierung des Verhaltens und der Struktur digitaler Systeme. Erlernen der Fähigkeit zur Spezifikation, Optimierung und Realisierung digitaler Systeme auf verschiedenen Abstraktionsebenen, einschließlich der Register-Transfer-Ebene. Verständnis der wichtigsten strukturellen und operationellen Eigenschaften eines Prozessors bis hin zur Schnittstelle mit der Software.

Turnus: Jährlich, Beginn in jedem Sommersemester

Voraussetzungen für die Teilnahme: Modul PRG

Prüfungsvorleistungen: -

Modulprüfung, Prüfungsform: 120-minütige Klausur, benotet

Für die Prüfung verantwortlicher Fachbereich: Informatik**Voraussetzung für die Vergabe der Leistungspunkte des Moduls:** Bestehen der Modulabschlussprüfung**Verwendbarkeit:** Bachelor, Master, soweit nicht schon im Bachelor eingebracht

				Semester/CP	
Lehrveranstaltung	Turnus	Typ	SWS	Jedes WS	
Hardware Architektur und Rechensysteme	WS	V	3	8	-
		Ü	2		

Lehrveranstaltung:	Hardware Architektur und Rechensysteme
Inhalt:	<p>Die Vorlesung bietet eine Einführung in den Aufbau und Entwurf digitaler Systeme.</p> <p>In der Vorlesung werden zunächst grundlegende Charakterisierungen von Hardwaresystemen wie analog/digital, sequentiell/kombinatorisch und synchron/asynchron behandelt und anhand von Beispielen ein erster Einblick in typische Entwurfsstrategien wie top-down oder bottom up gewährt. Zur Motivation des gesamten digitalen Schaltungs- und Systementwurfs werden die Grundlagen der modernen Rechnerarchitektur an den Anfang der Vorlesung gestellt. Behandelt werden die grundlegenden Prinzipien der Hardware-System-Architekturen (HSA) moderner Mikroprozessoren. Es wird ein erster Einblick in die Abarbeitung von Befehlen in Prozessoren vermittelt. Beendet wird diese Einführung mit einigen Beispielen zur Assemblerprogrammierung. Die folgenden Kapitel der Vorlesung konzentrieren sich auf den systematischen Entwurf digitaler Schaltnetze und Schaltwerke. Hierzu wird in die Grundlagen der booleschen Algebra eingeführt. Die ursprüngliche Intention der Booleschen Algebra war die Beschreibung der Gesetzmäßigkeiten des menschlichen Denkens. [G. Boole 1854: "those operations of the mind by which reasoning is performed."] In der Vorlesung wird die wichtige Erkenntnis von Claude Shannon (1936) vermittelt, dass die Boolesche Algebra insbesondere auch wichtige Gesetzmäßigkeiten in Schalternetzwerken beschreibt und somit eine mächtige mathematische Grundlage für die Entwurfsmethodik moderner digitaler Systeme darstellt. Die Vorlesung vertieft den Umgang mit den Booleschen Gesetzen und wendet sie zur Optimierung von Schaltkreisen an. Der systematische Entwurf digitaler Schaltnetze (kombinatorische Schaltungen) befasst sich mit der Bedeutung verschiedener Darstellungsarten boolescher Funktionen, den Optimierungsstrategien einschließlich der zeitlichen Modellierung sowie des Entwurfs und der Analyse exemplarischer Schaltnetze in den Datenpfaden von Prozessoren. Der dritte Schwerpunkt der Vorlesung widmet sich dem Entwurf sequentieller Systeme. Dazu werden zunächst grundlegende Begriffe der Automatentheorie behandelt und anschließend die Vorgehensweise beim Entwurf sequentieller Schaltungen anhand praktischer Beispiele betrachtet. Von besonderer Bedeutung sind die Optimierung sequentieller Schaltungen auf der Basis von Zustandsreduktion, Zustandscodierung und Schaltnetzoptimierung. Die Grundlage des Schaltnetz- und Schaltwerksentwurfs münden in die Prozessormodellierung und den Prozessorentwurf auf Registertransferebene. Die Vorlesung schließt daher den Kreis zu den Organisationsprinzipien der Rechnerarchitektur, die zu Beginn der Vorlesung behandelt wurden. Den Abschluss bildet eine Einführung in eine Hardwarebeschreibungssprache. Dem grundlegenden Aspekt der Modellierung wird in allen Kapiteln der Vorlesung in besonderer Weise Rechnung getragen.</p>
Vorkenntnisse:	PRG1
Studiennachweis:	Regelmäßige und erfolgreiche Teilnahme an den Übungen

Prüfung:	Klausur zu Veranstaltung ist gleichzeitig die Modulabschlussprüfung (s.o.)
----------	--

Modul DS Datenstrukturen (5 CP)

Inhalt, Ziel: Kenntnis fundamentaler Datenstrukturen und Algorithmen, sowie Vertrautheit mit dem Entwurf und der Analyse von Datenstrukturen und Algorithmen.

Die Vorlesung behandelt die Laufzeitanalyse, fundamentale Datenstrukturen und Algorithmen, zum anderen und allgemeine Methoden für den Entwurf und die Analyse von Datenstrukturen. Die Analyse von Datenstrukturen im Hinblick auf Laufzeit und Speicherplatzbedarf wird motiviert. Die asymptotische Notation wird eingeführt, und Methoden zur Lösung von Rekursionsgleichungen werden besprochen. Weiter werden die Darstellung von Bäumen und allgemeinen Graphen im Rechner und Algorithmen zur systematischen Durchmusterung von Graphen diskutiert und die Lösung weiterer Graphenprobleme und Algorithmen. Eine integrierte Sicht wird angelegt: Die vier genannten Aspekte werden eng verzahnt beschrieben: Algorithmen greifen auf Datenstrukturen zurück, Datenstrukturen benötigen Zugriffsalgorithmen, und der Entwurf von Datenstrukturen und Algorithmen wird durch deren Analyse mit geleitet und validiert. Die Analyse von Algorithmen im Hinblick auf Laufzeit und Speicherplatzbedarf wird motiviert. Die asymptotische Notation wird eingeführt, und Methoden zur Lösung von Rekursionsgleichungen werden besprochen. Elementare Datenstrukturen wie Listen, Keller und Warteschlangen werden beschrieben und analysiert. Der Begriff des abstrakten Datentyps wird eingeführt und motiviert, und effiziente Realisierungen der Datentypen des Wörterbuchs und der Prioritätswarteschlange unter Benutzung von Bäumen (beispielsweise AVL- und Splay-Bäume) und Hashing (auch verteiltes Hashing und Bloom-Filter) werden besprochen. Außerdem werden effiziente Datenstrukturen für das Union-Find-Problem behandelt. Algorithmen für Ordnungsprobleme wie Sortieren und Mischen werden beschrieben. Weiter wird die Darstellung von Bäumen und allgemeinen Graphen im Rechner diskutiert, und Algorithmen zur systematischen Durchmusterung von Graphen sowie zur Lösung weiterer Graphenprobleme wie die Berechnung kürzester Wege und minimaler Spannbäume werden beschrieben und analysiert. Es werden Beispiele für Algorithmentypen bzw. Entwurfsmethoden wie Greedy-Algorithmen, Teile-und-Beherrsche und dynamisches Programmieren angeführt, und ein Ausblick auf die Behandlung komplexer algorithmischer Probleme wird gegeben. Die um einen Ausblick auf weitere Methoden des Algorithmenentwurfs ergänzt werden.

Turnus: Jährlich, Beginn in jedem Sommersemester

Voraussetzungen für die Teilnahme: Modul PRG

Prüfungsvorleistungen: -

Modulprüfung, Prüfungsform: 100-minütige Klausur, benotet

Für die Prüfung verantwortlicher Fachbereich: Informatik

Voraussetzung für die Vergabe der Leistungspunkte des Moduls: Bestehen der Modulabschlussprüfung

Verwendbarkeit: Bachelor, Master, soweit nicht schon im Bachelor eingebracht

					Semester/CP
Lehrveranstaltung	Turnus	Typ	SWS	Jedes SS	
Hardware Architektur und Rechensysteme	WS	V Ü	2 1	5	-
Lehrveranstaltung:	Datenstrukturen				
Inhalt:	Datenstrukturen und Algorithmen				
Vorkenntnisse:	PRG1				
Studiennachweis:	Regelmäßige und erfolgreiche Teilnahme an den Übungen				
Prüfung:	Klausur zu Veranstaltung ist gleichzeitig die Modulabschlussprüfung (s.o.)				

Modul DM Diskrete Modellierung (7 CP)

Inhalt, Ziel: Kenntnis der grundlegenden Modellierungsmethoden und Beherrschen der entsprechenden Techniken. Fähigkeit zur präzisen und formalen Ausdrucksweise bei der Analyse von Problemen. Die Teilnehmer sollen einen Überblick über grundlegende Modellierungsmethoden und Kalküle bekommen und dabei lernen, erste Erfahrungen an typischen Beispielen sammeln und die für die Methoden typischen Techniken erlernen. Insgesamt sollen sie lernen, sich bei der Analyse von Problemen präzise und deklarativ auszudrücken, die Scheu vor formalen Kalkülen verlieren und den praktischen Wert von präzisen Beschreibungen erkennen

Turnus: Jährlich, Beginn in jedem Sommersemester

Voraussetzungen für die Teilnahme: Modul PRG

Prüfungsvorleistungen: -

Modulprüfung, Prüfungsform: 100-minütige Klausur, benotet

Für die Prüfung verantwortlicher Fachbereich: Informatik

Voraussetzung für die Vergabe der Leistungspunkte des Moduls: Bestehen der Modulabschlussprüfung

Verwendbarkeit: Bachelor. Master, soweit nicht schon im Bachelor eingebracht

					Semester/CP
Lehrveranstaltung	Turnus	Typ	SWS	Jedes SS	
Diskrete Modellierung	WS	V Ü	3 2	7	-
Lehrveranstaltung:	Diskrete Modellierung				
Inhalt:	In der Informatik wird das Modellieren mittels diskreter Strukturen als typische Arbeitsmethode in vielen Bereichen angewandt. Es dient der präzisen Beschreibung von Problemen durch spezielle Modelle und ist damit Voraussetzung für die Lösung eines Problems bzw. ermöglicht oft einen systematischen Entwurf. In den verschiedenen Gebieten der Informatik werden unterschiedliche, jeweils an die Art der Probleme und Aufgaben angepasste, diskrete Modellierungsmethoden verwendet. Innerhalb der Veranstaltung sollen zunächst die grundlegenden Begriffe, wie z.B. „Modell“ und „Modellierung“, geklärt werden. Anschließend werden verschiedene Ausdrucksmittel der Modellierung untersucht: Grundlegende Kalküle, Aussagen- und Prädikatenlogik, Graphen, endliche Automaten, Markov-Ketten, kontextfreie Grammatiken, Kellerautomaten, kontextsensitive Grammatiken, Entity-Relationship-Modell, Petri-Netze				
Vorkenntnisse:	PRG1				
Studiennachweis:	Regelmäßige und erfolgreiche Teilnahme an den Übungen				
Prüfung:	Klausur zu Veranstaltung ist gleichzeitig die Modulabschlussprüfung (s.o.)				