

Modulhandbuch

B.Sc. Medizinische Informatik

Fakultät für Angewandte Informatik

Wintersemester 2018/2019

Studienbeginn ab Wintersemester 2018/19

Liebe Studierenden,

willkommen im B.Sc. Medizinische Informatik an der Universität Augsburg!

Euer Studiengang ist komplett neu und findet in Zusammenarbeit der Fakultät für Angewandte Informatik mit der neuen Medizinischen Fakultät statt. Da die Medizinische Fakultät noch ganz am Anfang steht und euer Studiengang Inhalte beider Fakultäten verbindet, findet ihr in eurem Modulhandbuch für das WiSe 2018/2019 erstmal nur bereits etablierte Module, d.h. Informatik-Grundlagen-Module und könnt auch nur diese einbringen.

In den nächsten Semestern wird das Modulhandbuch nach und nach erweitert und vsl. ab dem WiSe 2019/2020 werdet ihr auch die ersten medizinischen Module belegen können.

Zum Start schaut euch am besten den Stundenplan an, den ihr auf folgender Seite findet:

https://www.informatik.uni-augsburg.de/studium/studiengaenge/bachelor_mis.html

Darin findet ihr jene Lehrveranstaltungen, die ihr im ersten Semester besuchen solltet (zu den Vorlesungen jeweils nur eine Übung, Zuteilungsverfahren werden am ersten Vorlesungstermin erklärt!).

Außerdem findet ihr auf dieser Homepage weitere Informationen, wie z.B. zur Einführungsveranstaltung am 22. Oktober 2018 im Klinikum Augsburg.

Da das Modulhandbuch ein Service für euch als Studierende ist, arbeiten wir eng mit der Studierendenvertretung Informatik zusammen. Solltet Ihr Anregungen, Fragen, Kritik oder Verbesserungsvorschläge zum neuen Modulhandbuch haben, so teilt diese einfach der Studierendenvertretung Informatik mit. Ihr erreicht sie unter fsinfo@informatik.uni-augsburg.de und persönlich im Raum 1007N.

Bei spezifischen Fragen zu eurem Studiengang könnt ihr euch auch an euren Fachstudienberater Adrian Rumpold wenden: adrian.rumpold@informatik.uni-augsburg.de

Guten Start ins Studium und

viele Grüße,

Eure Modulhandbuch-Beauftragten

Martin Frieb und Adrian Rumpold

ID	Modul	Semester	ECTS	SWS	Prüfung
1	Informatik				
INF-0073	Datenbanksysteme	jedes Wintersemester	8	4 Vorlesung 2 Übung	Klausur 90Minuten
INF-0097	Informatik 1	jedes Wintersemester	8	4 Vorlesung 2 Übung	Klausur 120Minuten
INF-0098	Informatik 2	jedes Sommersemester	8	4 Vorlesung 2 Übung	Klausur 120Minuten
INF-0100	Programmierkurs	jedes Semester	4	2 Vorlesung 1 Übung	praktische Prüfung 150Minuten
INF-0110	Einführung in die Theoretische Informatik	jedes Sommersemester	8	4 Vorlesung 2 Übung	Klausur 120Minuten
INF-0111	Informatik 3	jedes Wintersemester	8	4 Vorlesung 2 Übung	Klausur 120Minuten
INF-0120	Softwaretechnik	jedes Wintersemester	8	2 Vorlesung 4 Übung	Klausur 90Minuten

ID	Modul	Semester	ECTS	SWS	Prüfung
2	Mathematik				
MTH-6000	Mathematik für Informatiker I	jedes Wintersemester	8	4 Vorlesung 2 Übung 2 Übung 2 Vorlesung + Übung	Klausur 180Minuten
MTH-6010	Mathematik für Informatiker II	jedes Sommersemester	8	4 Vorlesung 2 Übung 2 Übung 2 Übung	Klausur 180Minuten

ID	Modul	Semester	ECTS	SWS	Prüfung
4	Medizinische Grundlagen -- Seminar				

ID	Modul	Semester	ECTS	SWS	Prüfung
7	Bachelorarbeit				
INF-0005	Bachelorarbeit	jedes Semester	12	0	Bachelorarbeit

Übersicht nach Modulgruppen

1) Informatik ECTS: 52

INF-0097: Informatik 1 (8 ECTS/LP, Pflicht) *	3
INF-0100: Programmierkurs (4 ECTS/LP, Pflicht) *	5
INF-0073: Datenbanksysteme (8 ECTS/LP, Pflicht) *	7
INF-0098: Informatik 2 (8 ECTS/LP, Pflicht)	9
INF-0110: Einführung in die Theoretische Informatik (8 ECTS/LP, Pflicht)	11
INF-0111: Informatik 3 (8 ECTS/LP, Pflicht) *	12
INF-0120: Softwaretechnik (8 ECTS/LP, Pflicht) *	14

2) Mathematik ECTS: 21

MTH-6000: Mathematik für Informatiker I (8 ECTS/LP, Pflicht) *	16
MTH-6010: Mathematik für Informatiker II (8 ECTS/LP, Pflicht)	18

3) Medizinische Grundlagen -- Seminar ECTS: 4

4) Bachelorarbeit ECTS: 12

INF-0005: Bachelorarbeit (12 ECTS/LP, Pflicht)	20
--	----

Modul INF-0097: Informatik 1 <i>Computer Science 1</i>		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Robert Lorenz		
<p>Lernziele/Kompetenzen: Teilnehmer verstehen die folgenden wesentlichen Konzepte der Informatik auf einem grundlegenden, Praxisorientierten, aber wissenschaftlichen Niveau: Architektur und Funktionsweise von Rechnern, Informationsdarstellung, Problemspezifikation, Algorithmus, Programm, Datenstruktur, Programmiersprache. Sie können einfache algorithmische Problemstellungen unter Bewertung verschiedener Entwurfsalternativen durch Programmiersprachenunabhängige Modelle lösen und diese in C oder einer ähnlichen imperativen Sprache implementieren. Sie können einfache Kommandozeilen-Anwendungen unter Auswahl geeigneter, ggf. auch dynamischer, Datenstrukturen durch ein geeignet in mehrere Übersetzungseinheiten strukturiertes C-Programm implementieren. Sie verstehen die imperativen Programmiersprachen zugrundeliegenden Konzepte und Modelle und sind in der Lage, andere imperative Programmiersprachen eigenständig zu erlernen. Sie kennen elementare Techniken zur Verifizierung und zur Berechnung der Komplexität von imperativen Programmen und können diese auf einfache Programme anwenden.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständiges Arbeiten mit Lehrbüchern; Eigenständiges Arbeiten mit Programmbibliotheken; Verständliche Präsentation von Ergebnissen; Fertigkeit der Zusammenarbeit in Teams</p>		
<p>Bemerkung: Dieses Modul entspricht der Veranstaltung "Einführung in die Informatik" für Wirtschaftsinformatiker</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p>		
<p>Voraussetzungen: keine</p>		
<p>Angebotshäufigkeit: jedes Wintersemester</p>	<p>Empfohlenes Fachsemester: 1.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 6</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Moduleile</p>		
<p>Moduleil: Informatik 1 (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 4</p>		

Inhalte:

In dieser Vorlesung wird als Einstieg in die praktische Informatik vermittelt, wie man Probleme der Informationsspeicherung und Informationsverarbeitung mit dem Rechner löst, angefangen bei der Formulierung einer Problemstellung, über den Entwurf eines Algorithmus bis zur Implementierung eines Programms. Die Vorlesung bietet eine Einführung in folgende Themenbereiche:

1. Rechnerarchitektur
2. Informationsdarstellung
3. Betriebssystem
4. Der Begriff des Algorithmus (Definition, Darstellung, Determinismus, Rekursion, Korrektheit, Effizienz)
5. Datenstruktur
6. Programmiersprache
7. Programmieren in C

Literatur:

- R. Richter, P. Sander und W. Stucky: Problem, Algorithmus, Programm , Teubner
- R. Richter, P. Sander und W. Stucky: Der Rechner als System, Teubner
- H. Erlenkötter: C Programmieren von Anfang an, rororo, 2008
- Gumm, Sommer: Einführung in die Informatik
- B. W. Kernighan, D. M. Ritchie, A.-T. Schreiner und E. Janich: Programmieren in C, Hanser
- C Standard Bibliothek: <http://www2.hs-fulda.de/~klingebiel/c-stdlib/>
- The GNU C Library: http://www.gnu.org/software/libc/manual/html_mono/libc.html

Zugeordnete Lehrveranstaltungen:

Informatik 1 (Vorlesung)

In dieser Vorlesung wird als Einstieg in die praktische Informatik vermittelt, wie man Probleme der Informationsspeicherung und Informationsverarbeitung mit dem Rechner löst, angefangen bei der Formulierung einer Problemstellung, über den Entwurf eines Algorithmus bis zur Implementierung eines Programms. Die Vorlesung bietet eine Einführung in folgende Themenbereiche: 1. Rechnerarchitektur 2. Informationsdarstellung 3. Betriebssystem 4. Der Begriff des Algorithmus (Definition, Darstellung, Rekursion, Korrektheit, Effizienz) 5. Datenstrukturen 6. Programmiersprachen 7. Programmieren in C Diese Vorlesung ist Voraussetzung für alle weiteren Veranstaltungen.

Modulteil: Informatik 1 (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Informatik 1 (Übung)

Prüfung

Informatik 1 (Klausur)

Klausur / Prüfungsdauer: 120 Minuten

Beschreibung:

Die Prüfung findet in der Regel in der 3. Woche nach Vorlesungsende (Ende Februar / Anfang März)) statt. Sie kann im darauf folgenden Semester vor Beginn der Vorlesungszeit (Anfang April) wiederholt werden.

Modul INF-0100: Programmierkurs		4 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Robert Lorenz		
<p>Lernziele/Kompetenzen: Teilnehmer verstehen die der verwendeten Programmiersprache zugrundeliegenden Konzepte und Modelle, kennen spezifische Entwurfstechniken und Methoden des strukturierten Programmierens und können diese auf praktisch relevante Problemstellungen mittlerer Größe anwenden. Sie beherrschen den Umgang mit Entwicklungsumgebungen und können sich selbstständig in Programmbibliotheken einarbeiten.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständiges Arbeiten mit Programmbibliotheken; Fertigkeit der Zusammenarbeit in Teams.</p>		
<p>Bemerkung: Der Programmierkurs wird entweder im ersten Semester in C aufbauend auf der Vorlesung "Informatik 1" oder im zweiten Semester in Java aufbauend auf der Vorlesung "Informatik 2" angeboten. Er findet jeweils als 1-wöchige Blockveranstaltung gegen Ende des Semesters statt (Wintersemester: Ende März / Sommersemester: Ende September).</p>		
<p>Arbeitsaufwand: Gesamt: 120 Std. 30 Std. Vorlesung (Präsenzstudium) 15 Std. Übung (Präsenzstudium) 15 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 15 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 45 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)</p>		
<p>Voraussetzungen: Grundlegende Kenntnisse in den Programmiersprachen C (C-Kurs) bzw. Java (Java-Kurs) Modul Informatik 1 (INF-0097) - empfohlen Modul Informatik 2 (INF-0098) - empfohlen</p>		
Angebotshäufigkeit: jedes Semester	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 3	Wiederholbarkeit: siehe PO des Studiengangs	

Moduleile
<p>Modulteil: Programmierkurs (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch / Englisch SWS: 2</p>
<p>Inhalte: Der Programmierkurs wird in den beiden Programmierspachen C und Java angeboten. Es werden anhand praktisch relevanter Problemstellungen die in Informatik 1 (Programmiersprache C) bzw. Informatik 2 (Java) erworbenen Programmierkenntnisse fachspezifisch vertieft.</p> <p>Themenauswahl:</p> <ul style="list-style-type: none"> • Mathematische Verfahren, • Dateien-Eingabe und -Ausgabe, • Grafische Simulationen, • Netzwerk-Kommunikation

Literatur:

- Programmiersprache C: B. W. Kernighan, D. M. Ritchie, A.-T. Schreiner und E. Janich: Programmieren in C, Hanser
- C Standard Bibliothek: <http://www2.hs-fulda.de/~klingebiel/c-stdlib/>
- The GNU C Library: http://www.gnu.org/software/libc/manual/html_mono/libc.html
- Ch. Ullenboom, Java ist auch eine Insel, Galileo Computing, <http://openbook.galileocomputing.de/javainsel/>
- Ch. Ullenboom, Mehr als eine Insel, Galileo Computing, <http://openbook.galileocomputing.de/java7/>
- M. Campione und K. Walrath, Das Java Tutorial, Addison Wesley, <http://docs.oracle.com/javase/tutorial/>
- Java-Dokumentation: <http://docs.oracle.com/javase/8/docs/api/>

Zugeordnete Lehrveranstaltungen:

Programmierkurs (Vorlesung)

C-Programmierkurs: In dem einwöchigen Kurs werden in Teamarbeit einige komplexere Problemstellungen unter Verwendung der Programmiersprache C bearbeitet und die in Informatik 1 erworbenen

Programmierkenntnisse vertieft. Mögliche Inhalte: - Strukturierte Vorgehensweise beim Erstellen von Programmen - Problemlösungsstrategien anwenden (Backtracking, Divide and Conquer, Branch and Bound, Problemtransformation, ...) - Dokumentation der Standard-Bibliothek verstehen und anwenden - Praktische Problemstellungen durch C-Programme lösen (Algorithmen aus der Mathematik, z.B. für Approximation, Optimierung, Ver- und Entschlüsselung, oder Analyse von Daten in Dateien, dauerhafte Datenspeicherung,...)

Java-Programmierkurs: In dem einwöchigen Kurs werden in Teamarbeit einige komplexere Entwurfsmuster und praktisch relevante Problemstellungen unter Verwendung der Programmiersprache Java bearbeitet und die in Informatik 2 erworbenen Programmierkenntnisse vertieft. - Netzwerk-Kommunikation (z.B. E-Mail

... (weiter siehe Digicampus)

Modulteil: Programmierkurs (Übung)

Lehrformen: Übung

Sprache: Deutsch / Englisch

SWS: 1

Zugeordnete Lehrveranstaltungen:

Übung zu Programmierkurs (Übung)

Die Verwaltung der Übung erfolgt über den Digicampus-Kurs zur Vorlesung "Programmierkurs".

Prüfung

Abnahme von Programmieraufgaben

praktische Prüfung / Prüfungsdauer: 150 Minuten

Modul INF-0073: Datenbanksysteme		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Peter Michael Fischer		
<p>Lernziele/Kompetenzen: Nach der Teilnahme an der Veranstaltung sind die Studierenden in der Lage, die in der Vorlesung Datenbanksysteme I vermittelten fachlichen Grundlagen in die Praxis umzusetzen. Diese umfassen vor allem Datenorganisation, Datenmodelle, konzeptionelle Modellierung mit ER, das relationales Modell sowie deklarative Datendefinition und Anfragen mit SQL. Darüber hinaus haben die Studierenden ein grundlegendes Verständnis von Implementierungstechniken von Datenbanksystemen wie Datenspeicherung und Indexe, Anfragebearbeitung mit Optimierung und Transaktionsverwaltung und können deren Auswirkungen auf die Praxis einordnen.</p> <p>Sie verfügen über fachspezifische Kenntnisse grundlegende Problemstellungen im Bereich Datenbanken zu verstehen und durch Anwenden erlernter Fähigkeiten zu lösen.</p> <p>Schlüsselqualifikationen: Eigenständiges Arbeiten mit Lehrbüchern; Eigenständiges Arbeiten mit Datenbanksystemen; Abstraktionsfähigkeit; Analytische und strukturierte Problemlösungsstrategien; Umsetzen fachlicher Lösungskonzepte in Programm und Modelle, Fertigkeit zur Analyse und Strukturierung komplexer Informatikproblemstellungen; Kenntnisse der Vor-/Nachteile von Entwurfsalternativen und Bewertung im jeweiligen Zusammenhang; Auswahl und sichere Anwendung geeigneter Methoden; Kenntnisse von praxisrelevanten Aufgabenstellungen; Fertigkeit zur Lösung von Problemen unter praxisnahen Randbedingungen;</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium)</p>		
<p>Voraussetzungen: Modul Informatik 2 (INF-0098) - empfohlen</p>		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Datenbanksysteme (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 4		
<p>Inhalte: Die Vorlesung beinhaltet grundlegende Konzepte von Datenbanksystemen und deren Anwendungen. Konkrete Inhalte sind: DB-Architektur, Entity-Relationship-Modell, Relationenmodell, Relationale Query-Sprachen, SQL, Algebraische Query-Optimierung, Implementierung der Relationenalgebra, Ablaufsteuerung paralleler Transaktionen, DB-Recovery und verteilte Transaktionen, Normalformtheorie.</p>		
<p>Literatur:</p> <ul style="list-style-type: none"> • W. Kießling, G. Köstler: Multimedia-Kurs Datenbanksysteme • R. Elmasri, S. Navathe: Fundamentals of Database Systems • A. Kemper, A. Eickler: Datenbanksysteme • J. Ullman: Principles of Database and Knowledge-Base Systems 		

Zugeordnete Lehrveranstaltungen:

Datenbanksysteme I (Vorlesung)

Modulteil: Datenbanksysteme (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Zugeordnete Lehrveranstaltungen:

Übung zu Datenbanksysteme I (Übung)

Prüfung

Datenbanksysteme (Klausur)

Klausur / Prüfungsdauer: 90 Minuten

Modul INF-0098: Informatik 2		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Robert Lorenz		
<p>Lernziele/Kompetenzen: Teilnehmer verstehen die folgenden wesentlichen Konzepte/Begriffe der Informatik auf einem grundlegenden, Praxis-orientierten, aber wissenschaftlichen Niveau: Softwareentwurf, Analyse- und Entwurfsmodell, UML, Objektorientierung, Entwurfsmuster, Grafische Benutzeroberfläche, Parallele Programmierung, persistente Datenhaltung, Datenbanken, XML, HTML. Sie können überschaubare nebenläufige Anwendungen mit grafischer Benutzerschnittstelle und persistenter Datenhaltung unter Berücksichtigung einfacher Entwurfsmuster, verschiedener Entwurfsalternativen und einer 3-Schichten-Architektur durch statische und dynamische UML-Diagramme aus verschiedenen Perspektiven modellieren und entsprechend der Diagramme in Java oder einer ähnlichen objektorientierten Sprache implementieren. Sie verstehen die diesen Programmiersprachen zugrundeliegenden Konzepte und Modelle und sind in der Lage, andere objektorientierte Programmiersprachen eigenständig zu erlernen.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken; Eigenständiges Arbeiten mit Lehrbüchern; Eigenständiges Arbeiten mit Programmbibliotheken; Verständliche Präsentation von Ergebnissen; Fertigkeit der Zusammenarbeit in Teams</p>		
<p>Bemerkung: Die erste Hälfte dieser Veranstaltung entspricht der Veranstaltung "Einführung in die Softwaretechnik" für Wirtschaftsinformatiker</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen: Empfohlen: Vorlesung "Informatik 1", Programmierkenntnisse in einer imperativen Programmiersprache (zum Beispiel C) Modul Informatik 1 (INF-0097) - empfohlen</p>		
<p>Angebotshäufigkeit: jedes Sommersemester</p>	<p>Empfohlenes Fachsemester: 2.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 6</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p> <p>Modulteil: Informatik 2 (Vorlesung) Lehrformen: Vorlesung Sprache: Deutsch SWS: 4</p>		

Inhalte:

Ziel der Vorlesung ist eine Einführung in die objektorientierte Entwicklung größerer Softwaresysteme, angefangen bei der Erstellung von Systemmodellen in UML bis zur Implementierung in einer objektorientierten Programmiersprache. Die Vorlesung bietet eine Einführung in folgende Themenbereiche:

1. Softwareentwurf
2. Analyse- und Entwurfsprozess
3. Schichten-Architektur
4. UML-Diagramme
5. Objektorientierte Programmierung (Vererbung, abstrakte Klassen und Schnittstellen, Polymorphie)
6. Entwurfsmuster und Klassenbibliotheken
7. Ausnahmebehandlung
8. Datenhaltungs-Konzepte
9. Grafische Benutzeroberflächen
10. Parallele Programmierung
11. Programmieren in Java
12. Datenbanken
13. XML
14. HTML

Literatur:

- Ch. Ullenboom, Java ist auch eine Insel, Galileo Computing, <http://openbook.galileocomputing.de/javainsel/>
- Ch. Ullenboom, Mehr als eine Insel, Galileo Computing, <http://openbook.galileocomputing.de/java7/>
- M. Campione und K. Walrath, Das Java Tutorial, Addison Wesley, <http://docs.oracle.com/javase/tutorial/>
- Java-Dokumentation: <http://docs.oracle.com/javase/8/docs/api/>
- Helmut Balzert, Lehrbuch Grundlagen der Informatik , Spektrum
- Heide Balzert, Lehrbuch der Objektmodellierung , Spektrum
- B. Oesterreich, Objektorientierte Softwareentwicklung , Oldenbourg

Modulteil: Informatik 2 (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 2

Prüfung

Informatik 2 (Klausur)

Klausur / Prüfungsdauer: 120 Minuten

Beschreibung:

Die Prüfung findet in der Regel in der 3. Woche nach Vorlesungsende (Anfang / Mitte August)) statt. Sie kann im darauf folgenden Semester vor Beginn der Vorlesungszeit (Anfang Oktober) wiederholt werden.

Modul INF-0110: Einführung in die Theoretische Informatik		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Bernhard Möller		
<p>Lernziele/Kompetenzen: Die Studierenden haben ein detailliertes Verständnis der Methoden zur formalen Beschreibung syntaktischer Strukturen sowie zu Fragen der prinzipiellen Berechenbarkeit. Hierzu zählen einerseits Endliche Automaten, Kellerautomaten und Turingmaschinen, andererseits reguläre, kontextfreie, kontextsensitive und unbeschränkte Chomsky-Grammatiken. Die Studierenden können diese Kenntnisse in konkreten Fragestellungen anwenden.</p> <p>Schlüsselqualifikationen: analytisch-methodische Kompetenz; Abwägen von Lösungsansätzen; Abstraktionsfähigkeit; Training des logischen Denkens; eigenständiges Arbeiten mit Lehrbüchern und englischsprachiger Fachliteratur; Grundsätze guter wissenschaftlicher Praxis</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
Voraussetzungen: Modul Diskrete Strukturen für Informatiker (INF-0109) - empfohlen		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: 2.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Einführung in die Theoretische Informatik (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 4		
Inhalte: Formale Sprachen, Grammatiken, Chomsky-Hierarchie, Regelsysteme, mathematische Maschinen (endliche Automaten, Kellerautomaten, Turingmaschinen)		
Literatur: <ul style="list-style-type: none"> • Eigenes Skriptum • U. Schöning: Theoretische Informatik- kurz gefasst, Spektrum 2008 • J. Hopcroft, R. Motwani, J. Ullman: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie, Pearson 2011 		
Modulteil: Einführung in die Theoretische Informatik (Übung)		
Lehrformen: Übung		
Sprache: Deutsch		
SWS: 2		
Prüfung		
Einführung in die Theoretische Informatik (Klausur) Klausur / Prüfungsdauer: 120 Minuten		

Modul INF-0111: Informatik 3		8 ECTS/LP
Version 1.0.0 (seit SoSe14) Modulverantwortliche/r: Prof. Dr. Bernhard Möller		
<p>Lernziele/Kompetenzen: Die Studierenden verfügen über ein grundlegendes Verständnis von Algorithmen und Datenstrukturen, unter anderem betreffend effiziente Sortier- und Suchverfahren sowie die geschickte Speicherung großer Datenmengen mit entsprechenden Zugriffsoperationen. Sie können dieses in konkreten Fragestellungen anwenden und haben ausgewählte Teile der vorgestellten Verfahren eigenständig programmiert.</p> <p>Schlüsselqualifikationen: analytisch-methodische Kompetenz; Abwägen von Lösungsansätzen; Abstraktionsfähigkeit; Training des logischen Denkens; eigenständiges Arbeiten mit Lehrbüchern und englischsprachiger Fachliteratur; Grundsätze guter wissenschaftlicher Praxis</p>		
<p>Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 60 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium)</p>		
<p>Voraussetzungen: Modul Informatik 1 (INF-0097) - empfohlen Modul Informatik 2 (INF-0098) - empfohlen Modul Diskrete Strukturen für Informatiker (INF-0109) - empfohlen</p>		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: 3.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Informatik 3 (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 4		
Inhalte: Effizienzbetrachtungen, Bäume, Sortierverfahren, Hashtabellen, Union-Find-Strukturen, Graphen, kürzeste Wege, Minimalgerüste, Greedy-Algorithmen, Backtracking, Tabellierung, amortisierte Komplexität, NP-Vollständigkeit		
Literatur: <ul style="list-style-type: none"> • Eigenes Skriptum • M. Weiss: Data Structures and Algorithm Analysis in Java, Pearson 2011 		
Zugeordnete Lehrveranstaltungen:		
Informatik III (Vorlesung)		
Modulteil: Informatik 3 (Übung)		
Lehrformen: Übung		
Sprache: Deutsch		
SWS: 2		
Zugeordnete Lehrveranstaltungen:		
Übung zu Informatik III (Übung)		

Prüfung

Informatik 3 (Klausur)

Klausur / Prüfungsdauer: 120 Minuten

Modul INF-0120: Softwaretechnik		8 ECTS/LP
Version 2.0.0 (seit WS16/17) Modulverantwortliche/r: Prof. Dr. Wolfgang Reif		
<p>Lernziele/Kompetenzen: Die Studierenden können einen fortgeschrittenen Softwareentwicklungsprozess zur Entwicklung komplexer Softwaresysteme anwenden und dafür Abstraktionen und Architekturen entwerfen. Sie können fachliche Lösungskonzepte in Programme umsetzen. Sie haben die Fertigkeit zur Analyse und Strukturierung von Anforderungen und Lösungsstrategien bei der Softwareentwicklung. Die Studierenden sind in der Lage, funktionale Anforderungen geeignet zu modellieren, beispielsweise mittels Use-Cases. Sie können geeignete Entwurfsalternativen, -muster und -methoden bewerten, auswählen und anwenden. Sie haben die Fertigkeit, Ideen und Konzepte mittels geeigneter Diagramme der UML zu dokumentieren und verständlich und überzeugend darzustellen. Sie kennen grundlegende Aufgaben und Techniken der Qualitätssicherung und können diese im Projektkontext einsetzen.</p> <p>Außerdem kennen die Studierenden praxisrelevanten Aufgabenstellungen und können diese bearbeiten.</p> <p>Schlüsselqualifikationen:</p> <ul style="list-style-type: none"> • Analytisch-methodische Kompetenz • Abwägen von Lösungsansätzen • Erwerb von Abstraktionsfähigkeiten • Erlernen des eigenständigen Arbeitens mit Lehrbüchern • Zusammenarbeit in Teams 		
<p>Arbeitsaufwand: Gesamt: 240 Std. 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium)</p>		
Voraussetzungen: Modul Softwareprojekt (INF-0122) - empfohlen		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: 5.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	
Moduleile		
Modulteil: Softwaretechnik (Vorlesung)		
Lehrformen: Vorlesung		
Sprache: Deutsch		
SWS: 2		
<p>Inhalte: Inhalt der Vorlesung ist ein Überblick über Methoden zur systematischen Entwicklung von Software, unter anderem der Unified Process (UP). Es werden die Unified Modelling Language (UML) und zugehörige Werkzeuge verwendet, die auch in die Übungen einbezogen werden.</p> <p>Die Veranstaltung behandelt den Softwarelebenszyklus, wichtige Aktivitäten der Softwareentwicklung (Analyse, Spezifikation, Design, Implementierung und Testen), die UML als Modellierungssprache, grundlegende Architekturmuster, GRASP und Design Patterns sowie Qualitätssicherung.</p>		

Literatur:

- Craig Larman: Applying UML and Patterns (3. Edition), Prentice Hall 2005
- Rupp, Hahn, Queins, Jeckle, Zengler: UML 2 glasklar (2. Auflage), Hanser 2005
- Gamma, Helm, Johnson, Vlissides: Design Patterns - Elements of Reusable Object-Oriented Software, Addison-Wesley 1995
- UML Spezifikation
- Folienhandout

Zugeordnete Lehrveranstaltungen:

Softwaretechnik (Vorlesung)

Die Vorlesung gibt einen Überblick über Methoden zur systematischen Entwicklung von Software, speziell den Unified Process (UP). Dabei verwenden wir die Unified Modelling Language (UML) und aktuelle Tools, die auch in die Übungen einbezogen werden. Behandelte Themen sind u.a.: * Der Softwarelebenszyklus * Der Unified Process * Wichtige Aktivitäten der Softwareentwicklung: Analyse, Spezifikation, Design, Implementierung, Wartung * UML als Modellierungssprache * GRASP und Design Patterns * Qualitätssicherung, Testen

Modulteil: Softwaretechnik (Übung)

Lehrformen: Übung

Sprache: Deutsch

SWS: 4

Zugeordnete Lehrveranstaltungen:

Übung zu Softwaretechnik (Übung)

Prüfung

Softwaretechnik Klausur

Klausur / Prüfungsdauer: 90 Minuten

Beschreibung:

Die Prüfung kann jedes Semester in der Prüfungszeit abgelegt werden.

Modul MTH-6000: Mathematik für Informatiker I		8 ECTS/LP
Version 1.1.0 Modulverantwortliche/r: apl. Prof. Dr. Dirk Hachenberger		
Bemerkung: Anstelle der Vorlesung Mathematik für Informatiker I kann die Vorlesung Lineare Algebra I eingebracht werden.		
Arbeitsaufwand: Gesamt: 240 Std. 60 Std. Vorlesung (Präsenzstudium) 30 Std. Übung (Präsenzstudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium)		
Voraussetzungen: Grundlagen der Schulmathematik		
Angebotshäufigkeit: jedes Wintersemester	Empfohlenes Fachsemester: 1.	Minimale Dauer des Moduls: 1 Semester
SWS: 6	Wiederholbarkeit: siehe PO des Studiengangs	

Modulteile**Modulteil: Mathematik für Informatiker I (Vorlesung)****Lehrformen:** Vorlesung**Dozenten:** apl. Prof. Dr. Dirk Hachenberger**Sprache:** Deutsch**SWS:** 4**Inhalte:**

- Grundbegriffe und Prinzipien zum Einstieg in die Mathematik
- Algebraische Grundstrukturen
- Elementare Zahlentheorie und einige Anwendungen
- Grundlagen der Linearen Algebra
- weitere algebraische Grundlagen und Zahlbereiche
- Ergänzung: Determinanten, charakteristisches Polynom

Zugeordnete Lehrveranstaltungen:**Mathematik für Informatiker I (Vorlesung)**

Inhalt: • Grundbegriffe und Prinzipien zum Einstieg in die Mathematik: Beweisprinzipien, vollständige Induktion, Abbildungen und Äquivalenzrelationen, Binomialkoeffizienten. • Algebraische Grundstrukturen: Von Monoiden zu Gruppen, von Ringen zu Körpern, von Vektorräumen zu Algebren. • Elementare Zahlentheorie und einige Anwendungen: Teilbarkeit, Zahldarstellung, Euklidischer Algorithmus, Restklassenringe, Prüfzeichen-Codierung, RSA-Public-Key-Cryptosystem. • Grundlagen der Linearen Algebra: Vektorräume, Matrizen, Lösen linearer Gleichungssysteme, Invertierbarkeit von Matrizen, Basen und Dimension, lineare Abbildungen. • weitere algebraische Grundlagen und Zahlbereiche: Komplexe Zahlen, Quaternionen, Polynome, Auswertung und Interpolation, Eigenwerte und Minimalpolynom von quadratischen Matrizen. Schlüsselqualifikationen: • Erweiterung und Festigung des mathematischen Schulwissens. • Schulung der logischen und strukturierten Denkweise. • Die Fähigkeit, grundlegende mathematische Aufgabe ... (weiter siehe Digicampus)

<p>Modulteil: Mathematik für Informatiker I (Klausurenkurs) Lehrformen: Vorlesung + Übung Dozenten: apl. Prof. Dr. Dirk Hachenberger Sprache: Deutsch Angebotshäufigkeit: jedes Sommersemester SWS: 2</p>
<p>Inhalte: Hierbei handelt es sich um ein vorlesungsunabhängiges Prüfungsmodul zur Mathematik für Informatiker I, das im Sommersemester angeboten wird.</p>
<p>Prüfung Mathematik für Informatiker I (Klausur) Klausur / Prüfungsdauer: 180 Minuten</p>
<p>Modulteile</p>
<p>Modulteil: Mathematik für Informatiker I (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 2</p>
<p>Zugeordnete Lehrveranstaltungen: Übung 01 zu Mathematik für Informatiker I (Übung) Zum Begriff Übung gehören generell die folgenden Aspekte: • Aufarbeitung der Inhalte der Vorlesung, • Anwendung der Inhalte auf konkrete Probleme, • Lernen, mathematische Sachverhalte zu formulieren, • Förderung des strukturierten Denkens, • Lernen, Fragen zu stellen und Dinge zu hinterfragen. Im Rahmen einer Anfängervorlesung kann auf die Wichtigkeit einer Übung daher nicht häufig genug hingewiesen werden. Organisatorisch werden die Übungen so durchgeführt, dass zunächst die gesamten Teilnehmer auf kleinere überschaubare Übungsgruppen aufgeteilt werden, die jeweils zweistündig (einmal pro Woche) stattfinden und von studentischen bzw. wissenschaftlichen Hilfskräften (Tutoren) geleitet werden. In den Übungsgruppen werden Aufgaben mit aktuellem Bezug zur Vorlesung unter Anleitung der Tutoren selbständig bearbeitet. Im Rahmen der Übungen wird weiterhin wöchentlich ein Hausaufgabenblatt herausgegeben, welches innerhalb einer Woche schriftlich zu bearbeiten und abzugeben ist; dieses Übungsb ... (weiter siehe Digicampus)</p>
<p>Modulteil: Mathematik für Informatiker I (Globalübung) Lehrformen: Übung Sprache: Deutsch SWS: 2</p>
<p>Inhalte: Die Globalübung dient der Ergänzung der Vorlesung. Hier werden die Lösungen zu den Hausaufgabenblättern besprochen, weitere Beispiele zum Vorlesungsstoff behandelt und dabei Überblicke über einzelne behandelte Themengebiete sowie Zusammenfassungen gegeben.</p>
<p>Zugeordnete Lehrveranstaltungen: Globalübung zu Mathematik für Informatiker I Die Globalübung dient der Ergänzung der Vorlesung. Hier werden die Lösungen zu den Hausaufgabenblättern besprochen, weitere Beispiele zum Vorlesungsstoff behandelt und dabei Überblicke über einzelne behandelte Themengebiete sowie Zusammenfassungen gegeben.</p>

Modul MTH-6010: Mathematik für Informatiker II		8 ECTS/LP
Version 1.0.0 Modulverantwortliche/r: apl. Prof. Dr. Dirk Hachenberger		
Bemerkung: Anstelle der Vorlesung Mathematik für Informatiker II kann die Vorlesung Analysis I eingebracht werden.		
Arbeitsaufwand: Gesamt: 240 Std. 30 Std. Übung (Präsenzstudium) 60 Std. Vorlesung (Präsenzstudium) 90 Std. Vor- und Nachbereitung des Stoffes Übung/Fallstudien (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes durch Literatur (Selbststudium) 30 Std. Vor- und Nachbereitung des Stoffes anhand bereitgestellter Unterlagen (Selbststudium)		
Voraussetzungen: Mathematik für Informatiker I Modul Mathematik für Informatiker I (MTH-6000) - Pflicht		
Angebotshäufigkeit: jedes Sommersemester	Empfohlenes Fachsemester: 2.	Minimale Dauer des Moduls: 1 Semester
SWS: 8	Wiederholbarkeit: siehe PO des Studiengangs	
Modulteile		
Modulteil: Mathematik für Informatiker II (Vorlesung) Lehrformen: Vorlesung Dozenten: apl. Prof. Dr. Dirk Hachenberger Sprache: Deutsch SWS: 4		
Lernziele: <ul style="list-style-type: none"> • Verständnis für die Axiomatik der reellen Zahlen, Abschätzungen. • Sicherer Überblick über die wichtigsten elementaren Funktionen. • Anwenden der Grenzwertsätze und Berechnung von Grenzwerten bei Folgen und Reihen sowie von Potenzreihen. • Analyse von differenzierbaren Funktionen und Anwenden der grundlegenden Integrationsregeln. Schlüsselqualifikationen: Erweiterung und Vertiefung der in "Mathematik für Informatiker I" gewonnenen Kenntnisse und Fähigkeiten.		
Inhalte: <ul style="list-style-type: none"> • <i>Aufbau der reellen Zahlen</i> • <i>Grundlagen der Analysis</i> • <i>Reihen und Potenzreihen</i> • <i>Stetige Funktionen</i> • <i>Differentialrechnung</i> • <i>Integralrechnung</i> 		
Modulteil: Mathematik für Informatiker II (Übung) Lehrformen: Übung Sprache: Deutsch SWS: 2		

Modulteil: Mathematik für Informatiker II (Globalübung)

Lehrformen: Übung

Sprache: Deutsch

Angebotshäufigkeit: jedes Wintersemester

SWS: 2

Inhalte:

Die Globalübung dient der Ergänzung der Vorlesung.

Modulteile

Modulteil: Mathematik für Informatiker II (Klausurenkurs)

Lehrformen: Übung

Dozenten: apl. Prof. Dr. Dirk Hachenberger

Sprache: Deutsch

Angebotshäufigkeit: jedes Wintersemester

SWS: 2

Inhalte:

Hierbei handelt es sich um ein vorlesungsunabhängiges Prüfungsmodul zur Mathematik für Informatiker II, das im Wintersemester angeboten wird.

Prüfung

Mathematik für Informatiker II (Klausur)

Klausur / Prüfungsdauer: 180 Minuten

Modul INF-0005: Bachelorarbeit		12 ECTS/LP
Version 1.0.0 (seit WS17/18) Modulverantwortliche/r: Professorinnen und Professoren, die Module für diesen Studiengang anbieten		
<p>Lernziele/Kompetenzen: Die Studierenden sind mit der wissenschaftlichen Methodik sowie Techniken der Literaturrecherche vertraut, sind in der Lage, unter Anleitung praktische oder theoretische Methoden zur Bearbeitung eines vorgegebenen Themas einzusetzen. Sie besitzen die Kompetenz, ein Problem der Informatik innerhalb einer vorgegebenen Frist weitgehend selbständig mit wissenschaftlichen Methoden zu bearbeiten sowie die Ergebnisse schriftlich und mündlich verständlich und überzeugend darzustellen.</p> <p>Außerdem verfügen die Studierenden über detailliertes und aktuelles Wissen auf dem Gebiet der Bachelorarbeit und können in Forschungs- oder Anwendungsprojekten auf diesem Gebiet aktiv mitarbeiten. Dazu haben sie die Fertigkeit zur Analyse und Strukturierung von Problemstellungen auf dem Gebiet der Bachelorarbeit und die Fertigkeit zur Entwicklung und Umsetzung von Lösungsstrategien. Sie kennen Vor- und Nachteile von Lösungsalternativen, können diese im jeweiligen Anwendungszusammenhang bewerten und geeignete Lösungsalternativen sicher auswählen. Sie verstehen weiterführende Konzepte, Methoden, Verfahren, Techniken und Technologien auf dem Gebiet der Bachelorarbeit und können dieses Wissen in Forschungs- oder Anwendungsprojekten einbringen. Die Studierenden haben die Fähigkeit, ihr Wissen auf dem Gebiet der Bachelorarbeit und verwandte Gebiete selbstständig zu erweitern. Darüber hinaus können die Studierenden grundlegende Strategien, Methoden und klare Vorstellungen entwickeln über ihre Ziele und Prioritäten, sie bewerten ihren persönlichen Arbeitsstil und schaffen eine effiziente Nutzung ihrer Ressourcen. Sie wenden Hilfsmittel und Techniken der Selbstorganisation an, die ihrem persönlichen Arbeitsstil entsprechen.</p> <p>Schlüsselqualifikationen: Fertigkeit zum logischen, analytischen und konzeptionellen Denken, Team- und Kommunikationsfähigkeit, Durchhaltevermögen, schriftliche und mündliche Darstellung eigener (praktischer oder theoretischer) Ergebnisse, Fertigkeit der effizienten Ressourcennutzung, Einschätzung der Relevanz eigener Ergebnisse, Grundsätze guter wissenschaftlicher Praxis, Fähigkeit zur Selbstreflexion und eines reflektierten Arbeitsstils, Fertigkeit der verständlichen, sicheren und überzeugenden (schriftlichen und mündlichen) Darstellung von eigenen und fremden (praktischen oder theoretischen) Ideen, Konzepten und Ergebnissen und deren Dokumentation, Qualitätsbewusstsein, Akribie</p>		
<p>Arbeitsaufwand: Gesamt: 360 Std. 360 Std. Anfertigen von schriftlichen Arbeiten (Selbststudium)</p>		
<p>Voraussetzungen: Empfohlene Veranstaltungen werden vom jeweiligen Betreuer bekanntgegeben. Es wird empfohlen, vorher ein Seminar abgeleistet zu haben.</p>		
<p>Angebotshäufigkeit: jedes Semester</p>	<p>Empfohlenes Fachsemester: 6.</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>
<p>SWS: 0</p>	<p>Wiederholbarkeit: siehe PO des Studiengangs</p>	
<p>Modulteile</p>		
<p>Modulteil: Bachelorarbeit Sprache: Deutsch</p>		
<p>Inhalte: Entsprechend dem gewählten Thema</p>		
<p>Literatur: Die Festlegung der Literatur erfolgt abhängig vom konkreten Thema der Arbeit in Absprache mit dem Betreuer.</p>		

Prüfung

Bachelorarbeit

Bachelorarbeit