

Modulhandbuch Bachelor Informatik (nach BPO'20)

Entwurf

Stand: Juli 2020

Mathematik 1: Logik und Algebra <i>Mathematics 1</i>							Modulkennziffer: IBGT-M1		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Mathematik und Theoretische Informatik									
Anzahl der SWS	V 4	UE 2	K 0	S 0	Prak. 0	Proj. 0	Σ 6	Kreditpunkte: 9	Turnus angeboten in jedem WiSe
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: keine (außer Schulmathematik bzw. Vorkurs Mathematik)									
Vorgesehenes Semester: 1. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> • Fähig sein, mathematische Notation zu verstehen und zu verwenden. • Im Stande sein, über mathematische Gegenstände und Sachverhalte zu kommunizieren. • Logisches Denken und Abstraktionsfähigkeit trainiert haben. • Mit den für die Informatik wichtigen Grundlagen der Logik, Mengentheorie, Kombinatorik, linearen Algebra und Geometrie vertraut sein, die elementaren Resultate aus diesen Gebieten kennen und sie anwenden können. • Beweise verstehen, nachvollziehen und selbständig durchführen können. 									
Inhalte: . <ol style="list-style-type: none"> 1 Logik: <ul style="list-style-type: none"> • Aussagen, Aussagenformen, Logische Operatoren, Normalformen • Logisches Schließen 2 Mengen: <ul style="list-style-type: none"> • Mengenbegriff, Mengenoperationen 3 Relationen und Abbildungen: <ul style="list-style-type: none"> • Äquivalenz – und Ordnungsrelationen • Abbildungen, Injektivität, Surjektivität, Bijektivität • Mächtigkeit von Mengen (N, Z, Q, R) 4 Vollständige Induktion: <ul style="list-style-type: none"> • Schwache und Starke Induktion 5 Kombinatorik: <ul style="list-style-type: none"> • Binomialkoeffizienten • Urnenmodell, Multinomialkoeffizienten 0,5 Inklusion-Exklusion • Schubfachprinzip • Bijektive Abbildungen und Permutationen 6 Algebra: <ul style="list-style-type: none"> • Gruppen, Untergruppen, Normateiler, Homomorphismen, Quotienten • Ringe (vor allem: Z), Polynome 1 Körper (R, Z_p) 									

Inhalte 2: .

7 Geometrie und lineare Algebra:

- Koordinaten und Basis, Standardbasis 0,5 Punkte, Geraden, Ebenen (in \mathbb{R}^3)
- Basiswechsel
- Lineare Gleichungssystem ($Ax=b$) 1 Rechnen mit Matrizen
- Einführung von Lineare Abbildungen
- Eigenwerte, Eigenräume
- Basiswechsel als lineare Abbildung/Ähnliche Matrizen
- Winkel und Skalarprodukt
- Singulärwertzerlegung

Lehrveranstaltung(en):

- 03-IBGT-M1 Mathematik 1: Logik, Kombinatorik und Lineare Algebra

Unterlagen (Skripte, Literatur, Programme usw.):

- G. und S. Teschl, Mathematik für Informatiker - Band 1: Diskrete Mathematik und Lineare Algebra. Springer 2006.
- P. Hartmann, Mathematik für Informatiker: ein praxisbezogenes Lehrbuch. Vieweg+Teubner, 5. Auflage 2012.
- E. Lehmann, F. Thomson Leighton, A.R. Meyer, Mathematics for computer science. MIT Skript 2011, Creative Commons (kostenlos online).
- W.Doerfler,W.Peschek: Einführung in die Mathematik für Informatiker. Hanser Verlag 1988
- Ch.Meinel,M.Mundhenk: Mathematische Grundlagen der Informatik, 2.Auflage, Teubner Verlag 2002.
- R.L.Graham,D.E.Knuth,O.Patashnik: Concrete Mathematics. A Foundation for Computer Science.Addison-Wesley Publ.Co.1988

Form der Prüfung:

KP, PL1: xx%, PL2: xx%, Portfolio, Klausur

Arbeitsaufwand	Präsenz	84 h
	Übungsbetrieb/Prüfungsvorbereitung	186 h
	Summe	270 h
Lehrende: SG Mathematik: Dr. T. Haga		Verantwortlich: Prof. Dr. C. Lutz

Mathematik 2 <i>Mathematics 2</i>							Modulkennziffer: IBGT-M2
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>							
Studienabschnitt: Grundlagen Mathematik und Theoretische Informatik							
Anzahl der SWS	V 3	UE 1	K 0	S 0	Prak. 0	Proj. 0	Σ 4
Kreditpunkte: 6						Turnus angeboten in jedem SoSe	
Formale Voraussetzungen: -							
Inhaltliche Voraussetzungen: Inhalte von Mathematik 1							
Vorgesehenes Semester: 2. Semester							
Sprache: Deutsch							
Ziele: <ul style="list-style-type: none"> • Fähig sein, mathematische Notation zu verstehen und zu verwenden. • Im Stande sein, über mathematische Gegenstände und Sachverhalte zu kommunizieren. • Logisches Denken und Abstraktionsfähigkeit trainiert haben. • Mit den für die Informatik wichtigen Grundlagen der Analysis, Differentialrechnung, Integralrechnung und Differentialgleichungen vertraut sein, die elementaren Resultate aus diesen Gebieten kennen und sie anwenden können. • Beweise verstehen, nachvollziehen und selbständig durchführen können. 							
Inhalte: . 1 Reelle und Komplexe Zahlen: <ul style="list-style-type: none"> • Reelle Zahlen, Supremum, Infimum • Reelle Exponential- und Logarithmusfunktion 1 Komplexe Zahlen • Komplexe Funktionen 2 Konvergenz: <ul style="list-style-type: none"> • Folgen, Reihen • Potenzreihen 3 Stetigkeit: <ul style="list-style-type: none"> • Folgenstetigkeit, Epsilon-Delta-Stetigkeit 4 Konvergenz: <ul style="list-style-type: none"> • Gleichmäßige Konvergenz 5 Differentialrechnung: <ul style="list-style-type: none"> • Funktionen einer Veränderlichen • Funktionen mehrerer Veränderlicher 6 Integralrechnung: <ul style="list-style-type: none"> • Funktionen einer Veränderlichen • Fouriertransformation 7 Differentialgleichungen: <ul style="list-style-type: none"> • Lineare ODEs 							
Inhalte 2: . Lehrveranstaltung(en): <ul style="list-style-type: none"> • 03-IBGT-M2 Mathematik 2: Analysis 							

Unterlagen (Skripte, Literatur, Programme usw.):

- W.Doerfler,W.Peschek: Einführung in die Mathematik für Informatiker. Hanser Verlag 1988
- Ch.Meinel,M.Mundhenk: Mathematische Grundlagen der Informatik, 2.Auflage, Teubner Verlag 2002.
- R.L.Graham,D.E.Knuth,O.Patashnik: Concrete Mathematics. A Foundation for Computer Science. Addison-Wesley Publ.Co.1988

Form der Prüfung:

MP, Klausur, mit Bonusprüfung

Arbeitsaufwand	Präsenz	56 h
	Übungsbetrieb/Prüfungsvorbereitung	124 h
	Summe	180 h

Lehrende:

SG Mathematik: Dr. T. Haga

Verantwortlich:

Prof. Dr. C. Lutz

Mathematik 3 <i>Mathematics 3</i>							Modulkennziffer: IBGT-M3		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Mathematik und Theoretische Informatik									
Anzahl der SWS	V 3	UE 1	K 0	S 0	Prak. 0	Proj. 0	Σ 4	Kreditpunkte: 6	Turnus angeboten in jedem SoSe
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: Inhalte von Mathematik 1									
Vorgesehenes Semester: 4. Semester									
Sprache: Deutsch									
<p>Ziele:</p> <ul style="list-style-type: none"> • Fähig sein, mathematische Notation zu verstehen und zu verwenden. • Im Stande sein, über mathematische Gegenstände und Sachverhalte zu kommunizieren. • Logisches Denken und Abstraktionsfähigkeit trainiert haben. • Mit den für die Informatik wichtigen Grundlagen der Wahrscheinlichkeitstheorie und Statistik vertraut sein, die elementaren Resultate aus diesen Gebieten kennen und sie anwenden können. • Beweise verstehen, nachvollziehen und selbständig durchführen können. 									
<p>Inhalte: .</p> <p>1 Zufall und Wahrscheinlichkeit:</p> <ul style="list-style-type: none"> • Bayesscher und frequentistischer Wahrscheinlichkeitsbegriff • Bedingte Wahrscheinlichkeiten Stochastische Unabhängigkeit <p>2 Diskrete Verteilungen:</p> <ul style="list-style-type: none"> • Laplace-Verteilung • Poisson-Verteilung • Binomialverteilung <p>3 Stetige Verteilungen:</p> <ul style="list-style-type: none"> • Normal- und Standardnormalverteilung 1 Student-t-Verteilung • Chi-Quadrat-Verteilung <p>4 Parameter:</p> <ul style="list-style-type: none"> • Erwartungswert, Standardabweichung, Varianz • Ungleichungen (Chebychev, Chernov, Markov) <p>5 Deskriptive Statistik:</p> <ul style="list-style-type: none"> • Skalentypen • Modus, Median, Mittelwert, empirische Varianz <p>6 Schließende Statistik:</p> <ul style="list-style-type: none"> • Punkt- und Intervallschätzung des Erwartungswerts • Varianzschätzer • Hypothesentests 									

Inhalte 2: .

Lehrveranstaltung(en):

- 03-IBGT-M3 Mathematik 3: Stochastik und Statistik

Unterlagen (Skripte, Literatur, Programme usw.):

- L. Fahrmeir, C. Heumann, R Künstler, I. Pigeot, G. Tutz: Statistik - Der Weg zur Datenanalyse, Springer, 2016.
- Hans-Otto Georgii: Stochastik. Einführung in die Wahrscheinlichkeitstheorie und Statistik, De Gruyter, 2015
- Klenke: Wahrscheinlichkeitstheorie, Springer, 2013

Form der Prüfung:

KP, PL1: xx%, PL2: xx%, Portfolio, Klausur

Arbeitsaufwand	Präsenz	56 h
	Übungsbetrieb/Prüfungsvorbereitung	124 h
	Summe	180 h

Lehrende:
SG Mathematik: N.N.

Verantwortlich:
Prof. Dr. C. Lutz

Theoretische Informatik 1 <i>Theoretical Computer Science 1</i>							Modulkennziffer: IBGT-TH11		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Mathematik und Theoretische Informatik									
Anzahl der SWS	V 4	UE 2	K 0	S 0	Prak. 0	Proj. 0	Σ 6	Kreditpunkte: 9	Turnus angeboten in jedem WiSe
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 3. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> • Formale Grundlagen und elementare Fragestellungen der Informatik kennen und die fundamentale Rolle der Theorie in der Informatik verstehen. • Konzepte zur formalen Beschreibung und Analyse von Informatiksystemen kennen. • Beherrschung der grundlegenden Methoden aus den Bereichen der Automatentheorie, formalen Sprachen und Algorithmen. • Beherrschung elementarer Beweistechniken und Beweise selbst durchführen können. • Probleme analysieren, von spezifischen Gegebenheiten abstrahieren und formale Modelle in mathematischen Definitionen darstellen können. • Algorithmen für diese Probleme kennen und auf neue Problemvarianten anwenden können. • Korrektheit von Algorithmen beweisen und Eigenschaften von Algorithmen analysieren können. • Eigenständig und in Gruppen Lösungsstrategien für formale Problemstellungen entwickeln können und Lösungen verständlich präsentieren. 									
Inhalte: . A) Automaten und formale Sprachen 1 Endliche Automaten und Reguläre Sprachen: <ul style="list-style-type: none"> • Definition und Grundbegriffe • Nichtdeterminismus • Nichterkennbarkeit von Sprachen und Pumping-Lemma • Abschlusseigenschaften • Potenz- und Produktautomat • Leerheits-, Wort- und Äquivalenzproblem • reguläre Ausdrücke • Minimale Automaten und Nerode-Rechtskongruenz • Rechtslineare Grammatiken 									

Inhalte 2: .

2 Kontextfreie Sprachen:

- Grammatiken und Chomsky-Hierarchie
- kontextfreie Grammatiken
- Chomsky Normalform
- Leerheits-, Wort- und Äquivalenzproblem
- CYK-Algorithmus
- Abschlusseigenschaften
- Pumping-Lemma
- Kellerautomaten

B) Algorithmentheorie

- Algorithmenbegriff
- Korrektheit und Komplexität von Algorithmen
- Suchen und Rekursionen
- Sortieren
- Graphen und elementare Graphenalgorithmen: Graphdurchläufe, MST und SP
- Algorithmen Paradigmen: Divide and Conquer, Greedy Algorithmen, Dynamische Programmierung

Lehrveranstaltung(en): Dieses Modul besteht aus zwei Veranstaltungen, jeweils im Format 2VL+1UE, die beide verpflichtend sind.

- 03-IBGT-AFS Automaten und formale Sprachen
- 03-IBGT-AT Algorithmentheorie

Unterlagen (Skripte, Literatur, Programme usw.):

- J.E. Hopcroft, R. Motwani, J.D. Ullman: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie, Pearson Studium 2011
- J.E. Hopcroft, R. Motwani, J.D. Ullman: Introduction to Automata Theory, Languages, and Computation (3rd edition). Pearson Education, 2014
- C. Lutz: Theoretische Informatik 1, Skript
- D. Kozen: Automata and Computability, Springer, 2007
- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein: Introduction to Algorithms, MIT Press, 2003
- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein: Algorithmen - Eine Einführung, Walter de Gruyter GmbH & Co KG, 2017
- Martin Dietzfelbinger, Kurt Mehlhorn, Peter Sanders: Algorithmen und Datenstrukturen: Die Grundwerkzeuge, Springer-Verlag, 2014
- T. Ottmann and P. Widmayer. Algorithmen und Datenstrukturen. Spektrum Verlag, 2002.

Form der Prüfung:

TP, PL1: 50%, PL2: 50%, Klausur, Fachgespräch, ggf. Bonusprüfung

Arbeitsaufwand	Präsenz	84 h
	Übungsbetrieb/Prüfungsvorbereitung	186 h
	Summe	270 h

Lehrende:
Prof. Dr. C. Lutz, Prof. Dr. N. Megow, Prof. Dr. S. Siebertz

Verantwortlich:
Prof. Dr. C. Lutz

Theoretische Informatik 2 <i>Theoretical Computer Science 2</i>							Modulkennziffer: IBGT-THI2		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Mathematik und Theoretische Informatik									
Anzahl der SWS	V 2	UE 2	K 0	S 0	Prak. 0	Proj. 0	Σ 4	Kreditpunkte: 6	Turnus angeboten in jedem SoSe
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 4.Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> • Formale Grundlagen und elementare Fragestellungen der Informatik kennen und die fundamentale Rolle der Theorie in der Informatik verstehen. • Konzepte zur formalen Beschreibung und Analyse von Informatiksystemen kennen. • Beherrschung der grundlegenden Methoden aus den Bereichen der Berechenbarkeit und Komplexität. • Beherrschung elementarer Beweistechniken und Beweise selbst durchführen können. • Probleme analysieren, von spezifischen Gegebenheiten abstrahieren und formale Modelle in mathematischen Definitionen darstellen können. • Algorithmen für diese Probleme kennen und auf neue Problemvarianten anwenden können. • Korrektheit von Algorithmen beweisen und Eigenschaften von Algorithmen analysieren können. • Eigenständig und in Gruppen Lösungsstrategien für formale Problemstellungen entwickeln können und Lösungen verständlich präsentieren. 									
Inhalte: . <p>1 Berechenbarkeit:</p> <ul style="list-style-type: none"> • Turingmaschinen • Linear beschränkte Automaten • Grammatiken der Typen 0 und 1, Abschlusseigenschaften • LOOP-Programme und WHILE-Programme • Primitiv rekursive Funktionen und -rekursive Funktionen • Unentscheidbarkeit • Unentscheidbare Probleme für Turingmaschinen • Satz von Rice • Postsches Korrespondenzproblem • Äquivalenzproblem kontextfreier Grammatiken • Semi-Entscheidbarkeit und Rekursive Aufzählbarkeit • Universelle Turingmaschinen • Reduktionen 									

Inhalte 2: .

2 Komplexität:

- Zeit- und Platzbeschränkte Turingmaschinen
- Komplexitätsklassen P, NP, PSpace, ExpTime
- P vs NP-Problem
- NP-Vollständigkeit
- NP-vollständige Probleme aus verschiedenen Gebieten
- Komplemente und coNP
- Approximation NP-harter Probleme
- Satz von Savitch

Lehrveranstaltung(en):

- 03-IBGT-THI2 Theoretische Informatik 2: Berechenbarkeit und Komplexität

Unterlagen (Skripte, Literatur, Programme usw.):

- J.E. Hopcroft, R. Motwani, J.D. Ullman: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie, Pearson Studium 2011
- J.E. Hopcroft, R. Motwani, J.D. Ullman: Introduction to Automata Theory, Languages, and Computation (3rd edition). Pearson Education, 2014
- C. Lutz: Theoretische Informatik, Skript 2. Teil

Form der Prüfung:

MP, Fachgespräch, Klausur, ggf. mit Bonusprüfung

Arbeitsaufwand	Präsenz	56 h
	Übungsbetrieb/Prüfungsvorbereitung	124 h
	Summe	180 h

Lehrende:

Prof. Dr. C. Lutz, Prof. Dr. S. Siebertz

Verantwortlich:

Prof. Dr. C. Lutz

Praktische Informatik 1 <i>Practical Computer Science 1</i>							Modulkennziffer: IBGP-PI1		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Praktische und Technische Informatik									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 9	Turnus angeboten in jedem WiSe
	4	0	0	0	4	0	8		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 1. Semester									
Sprache: Deutsch									
<p>Ziele:</p> <ul style="list-style-type: none"> • Grundlegende Informatikkonzepte wiedergeben und erklären können. • Konzepte einer imperativen Programmiersprache kennen, verstehen und anwenden können. • Anschauliche Sachverhalte im Modell der Objektorientierung ausdrücken können. • Einfache Algorithmen entwickeln und in Java umsetzen können. • Einfache in Java realisierte Algorithmen systematisch testen können. • Probleme in Teilprobleme zerlegen und diese Strukturierung mit Mitteln von Java umsetzen und aussagekräftig dokumentieren können. • Formale Syntaxbeschreibungen verstehen können. • Eine einfache Entwicklungsumgebung nutzen können. • LaTeX zur Erstellung einfacher Dokumente nutzen können. • Versionsverwaltungssysteme einsetzen können. • In Gruppen Probleme analysieren und gemeinsam Lösungsstrategien entwickeln und präsentieren können. <p>Die Vorlesungen Praktische Informatik 1 und 2 vermitteln essenzielles Grundwissen und Basisfähigkeiten, deren Beherrschung für nahezu jede vertiefte Beschäftigung mit Informatik – sowohl in der industriellen Anwendung, als auch in der Forschung – Voraussetzung ist.</p>									

Inhalte:

1. Basiswissen: von Neumannsche Rechnerorganisation – Grundlagen der Rechnerarchitektur – Programm und Prozess – Programmiersprachen – Compiler, Assembler, Loader, Linker, Interpreter, Laufzeitumgebungen, Betriebssysteme – Grafische Benutzungsschnittstellen
2. Datenstrukturen: Information und ihre Repräsentation – Datentypen und Typanalyse – Elementare und zusammengesetzte Datentypen – rekursive Datentypen – Kanonische Operationen auf den eingeführten Datenstrukturen
3. Programmierparadigmen: (1) Imperative und funktionale Programmierung, (2) Objektorientierte (imperative) Programmierung, (3) Sequenzielle Programme versus nebenläufige Programme
4. Grundkomponenten imperativer Programmiersprachen: Schnittstellen und Ein-/Ausgabe, Variablen und Zuweisungen, Kontrollstrukturen, Blöcke, Funktionen, Rekursion
5. Syntax und Semantik imperativer Programmiersprachen: Syntax und Methoden der Syntax-Spezifikation, reguläre Ausdrücke, (erweiterte) Backus-Naur-Form (E)BNF
6. Prinzipien der objektorientierten Programmierung: Geheimnisprinzip – Methoden – Operationen – Objekte – Klassen – Botschaften – Ereignisverarbeitung – Attribute – Vererbung – Polymorphismus – Überladung – Generische Datentypen
7. Umsetzung der Punkte 2.-6. mit Java – Illustration anhand einfacher Algorithmen
8. Programmdokumentation und zugehörige Hilfswerkzeuge, z.B. JavaDoc
9. Testen von Programmen und zugehörige Hilfswerkzeuge, z.B. JUnit
10. Grundlagen der Netzwerkkommunikation: IP-Adressen, DNS, TCP, UDP
11. Grundkonzepte der Entwicklung graphischer Oberflächen

Programmier-Praktikum: Programmentwicklung in Java – Realisierung einzelner, überschaubarer Programmieraufgaben

Lehrveranstaltung(en):

- 03-IBGP-PI1 Praktische Informatik 1: Imperative Programmierung und Objektorientierung

Unterlagen (Skripte, Literatur, Programme usw.):

- David J. Barnes, Michael Kölling: Java lernen mit BlueJ - Objects first - Eine Einführung in Java. Aktuelle Auflage. Pearson Studium.

Weitere Informationen (Beispielprogramme, Musterlösungen, im WWW verfügbare Literatur) sind auf der Web-Seite der Veranstaltung zu finden.

Form der Prüfung:

KP; PL1: 70%, PL2: 30%; Portfolio, Klausur

Arbeitsaufwand	Präsenz	112 h
	Übungsbetrieb/Prüfungsvorbereitung	158 h
	Summe	270 h

Lehrende:

Dr. T. Röfer, N.N.

Verantwortlich:

Prof. Dr. U. Bormann

Praktische Informatik 2 <i>Practical Computer Science 2</i>							Modulkennziffer: IBGP-PI2		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Praktische und Technische Informatik									
Anzahl der SWS	V 2	UE 2	K 0	S 0	Prak. 0	Proj. 0	Σ 4	Kreditpunkte: 6	Turnus angeboten in jedem SoSe
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: Praktische Informatik 1									
Vorgesehenes Semester: 2. Semester									
Sprache: Deutsch									
<p>Ziele:</p> <ul style="list-style-type: none"> • Typische Datenstrukturen identifizieren und problemadäquat einsetzen können. • Datenstrukturen und Algorithmen in Java umsetzen können. • Wesentliche Algorithmen der Informatik erklären, anwenden und modifizieren können. • Algorithmische Alternativen bezüglich der Eignung für ein Problem beurteilen können. • Grundbegriffe der formalen Verifikation erläutern können. • Die Komplexität von einfachen Algorithmen analysieren können. • Eine komplexe Entwicklungsumgebung nutzen können. • Generische und funktionale Konzepte in eigenen Programmen einsetzen können. • In Gruppen Probleme analysieren und gemeinsam Lösungsstrategien entwickeln und präsentieren können. <p>Die Vorlesungen Praktische Informatik 1 und 2 vermitteln essenzielles Grundwissen und Basisfähigkeiten, deren Beherrschung für nahezu jede vertiefte Beschäftigung mit Informatik – sowohl in der industriellen Anwendung, als auch in der Forschung – Voraussetzung ist.</p>									
<p>Inhalte:</p> <ol style="list-style-type: none"> 1. Algorithmen: Begriff des Algorithmus – Beschreibung von Algorithmen – Algorithmische Umsetzung kanonischer Operationen auf Datenstrukturen – Grundlegende Strategien: Greedy, Divide-and-Conquer, Backtracking, dynamische Programmierung 2. Komplexität von Algorithmen – $O(n)$-Notation und asymptotische Analyse 3. Suchen und Sortieren auf Arrays: Binäre Suche – Quicksort und weitere Sortieralgorithmen – Komplexitätsvergleiche 4. Mengen – Multimengen – Relationen – Funktionen: Datenstrukturen und Algorithmen zur Realisierung kanonischer Operationen (z.B. Mengenalgebra) 5. Listen – Stapel – Warteschlangen: Datenstrukturen zur Realisierung (Arrays versus Verkettung und dynamische Speicherallokation für Elemente), Algorithmen zur Realisierung kanonischer Operationen (Listentraversalion, Anfügen, Einfügen, Löschen, Suchen, Stack-Operationen, FIFO-Warteschlangenoperationen) 6. Bäume: Binäre Bäume, AVL-Bäume, Rot-Schwarz-Bäume, B-Bäume – Suchen, Einfügen, Löschen, Traversalion 7. Hashing: Hash-Array, Hashfunktion, Hash Buckets, offenes Hashing 8. Graphen: ungerichtete, gerichtete, gewichtete Graphen – Repräsentation durch Knoten- und Kantenlisten, durch Adjazenzmatrizen, Adjazenzlisten – Algorithmen auf Graphen: Breitensuche, Tiefensuche, kürzeste Wege auf gewichteten Graphen: Dijkstras Algorithmus, minimal aufspannende Bäume: Algorithmen von Prim et al. und Kruskal 9. Spezifikation von Programmen: Vor- und Nachbedingungen – Invarianten 10. Verifikation: Partielle und totale Korrektheit sequenzieller Programme – Formale Verifikation, z.B. Hoare Logik (Pre-/Postconditions) – Eigenschaftsbeweis durch Strukturelle Induktion <p>Lehrveranstaltung(en):</p> <ul style="list-style-type: none"> • 03-IBGP-PI2 Praktische Informatik 2: Algorithmen und Datenstrukturen 									

Unterlagen (Skripte, Literatur, Programme usw.):

- G. Saake und K.-U. Sattler: Algorithmen und Datenstrukturen. dpunkt.verlag, Heidelberg (2004)
- R. Schiedermeier: Programmieren mit Java. Pearson, München (2005)

Weitere Informationen (Beispielprogramme, Musterlösungen, im WWW verfügbare Literatur) sind auf der Web-Seite der Veranstaltung zu finden.

Form der Prüfung:

KP, PL1: 70%, PL2: 30%, Portfolio, Klausur

Arbeitsaufwand	Präsenz	56 h
	Übungsbetrieb/Prüfungsvorbereitung	124 h
	Summe	180 h

Lehrende:
Dr. T. Röfer, N.N.

Verantwortlich:
Prof. Dr. U. Bormann

Praktische Informatik 3 <i>Practical Computer Science 3</i>							Modulkennziffer: IBGP-PI3		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Praktische und Technische Informatik									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus angeboten in jedem WiSe
	2	2	0	0	0	0	4		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: Praktische Informatik 2									
Vorgesehenes Semester: 3. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> • Konzepte und typische Merkmale des funktionalen Programmierens kennen, verstehen und anwenden können. • Vertieftes Verständnis von Datenstrukturen und Algorithmen. • In Gruppen Probleme analysieren und gemeinsam Lösungsstrategien entwickeln und präsentieren können. Die Vorlesung Praktische Informatik 3 vermittelt essenzielles Grundwissen und Basisfähigkeiten, deren Beherrschung für nahezu jede vertiefte Beschäftigung mit Informatik Voraussetzung ist.									
Inhalte: <ol style="list-style-type: none"> 1. Grundlagen der funktionalen Programmierung: Rekursion – Definition von Funktionen durch rekursive Gleichungen und Mustervergleich (pattern matching) – Auswertung, Reduktion, Normalform – Funktionen höherer Ordnung, currying, Typkorrektheit und Typinferenz 2. Typen: Algebraische Datentypen – Typkonstruktoren – Typklassen – Polymorphie – Standarddatentypen (Listen, kartesische Produkte, Lifting) und Standardfunktionen darauf (fold, map, filter) – Listenkomprehension 3. Algorithmen und Datenstrukturen: Unendliche Listen (Ströme) – Bäume – Graphen – zyklische Datenstrukturen 4. Strukturierung und Spezifikation: Module – Schnittstellen (Interfaces) – Abstrakte Datentypen – Signaturen und Axiome 5. Theoretische Aspekte: Referentielle Transparenz – Lambda-Kalkül – Beweis durch Induktion 6. Fortgeschrittene Funktionale Programmierung: Funktionale I/O und zustandsbasierte Programme – Monaden Im Übungsbetrieb; Programmentwicklung in Haskell — Realisierung einzelner, überschaubarer Programmieraufgaben in kleinen Gruppen									
Lehrveranstaltung(en): <ul style="list-style-type: none"> • 03-IBGP-PI3 Praktische Informatik 3: Funktionale Programmierung 									
Unterlagen (Skripte, Literatur, Programme usw.): <ul style="list-style-type: none"> • Simon Thompson: Haskell - The Craft of Functional Programming, Addison-Wesley, 3. Auflage 2011. Weiteres Lehrmaterial ist auf der Webseite des Veranstaltung zu finden: <ul style="list-style-type: none"> • Folienkopien • Übungsaufgaben • Hinweise auf Quellen im WWW Das Haskell-System ghci ist frei verfügbare Software (für Linux, Windows und MacOS).									
Form der Prüfung: KP, PL1: xx%, PL2: xx%, Portfolio, Klausur									
Arbeitsaufwand	Präsenz		56 h						
	Übungsbetrieb/Prüfungsvorbereitung		124 h						
	Summe		180 h						

Lehrende:

Prof. Dr. C. Lüth, Dr. Th. Barkowsky

Verantwortlich:

Prof. Dr. C. Lüth

Technische Informatik 1 <i>Technical Computer Science 1</i>							Modulkennziffer: IBGP-T11		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Praktische und Technische Informatik									
Anzahl der SWS	V 4	UE 2	K 0	S 0	Prak. 0	Proj. 0	Σ 6	Kreditpunkte: 9	Turnus angeboten in jedem SoSe
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 2. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> • Grundlegende Konzepte moderner Rechner wiedergeben und erläutern können • Schaltkreismodellierung durch Boolesche Funktionen verstehen und erklären können • Hardware-Realisierungen von arithmetischen Funktionen darstellen können • Modellierung und Optimierungsansätze integrierter Schaltkreise umreißen können • Rechnersysteme anhand der eingeführten Konzepte selbständig beurteilen können • Unterschiedliche Hardware-Realisierungen unter den eingeführten Optimierungskriterien bewerten können • In Gruppen Probleme analysieren, gemeinsam Lösungsstrategien entwickeln und präsentieren können 									
Inhalte: . I. Rechnerarchitektur <ol style="list-style-type: none"> 1. Rechnersichtweisen: Ebenen und Sprachen, Hierarchie, Compiler, Interpreter 2. Aufbau und Funktionsweise: Hardware, Software, Firmware, Aufbau eines von-Neumann-Rechners, Arbeitsspeicher, Speicherzelle, Arbeitsweise eines Prozessors, Speicher, I/O Busse 3. Befehlssatz: RISC, CISC, Designprinzipien 4. Pipelining 5. Speicher: Hierarchie, Organisation, Caches, Hintergrundspeicher 6. Parallelität: Ausprägungen, Klassifikation von parallelen Rechnerarchitekturen, Exkurs über Verbindungsstrukturen II. Digitale Schaltungen: <ol style="list-style-type: none"> 1. Schaltkreise: Technologien, Definition, Kosten, Semantik von kombinatorischen Schaltkreisen, Simulation, Teilschaltkreise, Hierarchischer Entwurf, Beispiele 2. Kodierung: Zeichen, Zahlen, Zahlensysteme, Übertragung, Fehlerkorrektur, HammingCode, Huffman-Code, Festkommadarstellungen, Zahlendarstellung durch Betrag und Vorzeichen, Einer-/Zweierkomplement-Darstellung, Gleitkommadarstellung (IEEE-754 Format) 3. Boolescher Kalkül: Funktion, Algebra, Ausdrücke, alternative Funktionsdarstellung, z.B. durch Entscheidungsdiagramme 4. Zweistufige Schaltungen: Logiksynthese, Implikanten, Primimplikanten, Minimierung, Quine/McClusky, Überdeckungsproblem 5. Integrierte Schaltungen, arithmetische Schaltungen, ALU 6. Schaltungen mit speichernden Elementen Lehrveranstaltung(en): <ul style="list-style-type: none"> • 03-IBGP-T11 Technische Informatik 1: Rechnerarchitektur und digitale Schaltungen 									

Unterlagen (Skripte, Literatur, Programme usw.): • B. Becker, R. Drechsler, P. Molitor: Technische Informatik – Eine Einführung, Pearson Studium, 2005

- B. Becker, P. Molitor: Technische Informatik - Eine einführende Darstellung, Oldenbourg Wissenschaftsverlag, 2008
- D. Hoffmann: Grundlagen der Technischen Informatik, 5. Aufl., Hanser Verlag, 2016
- A. S. Tanenbaum, T. Austin: Computerarchitektur, 6. Aufl., Pearson Studium, 2014
- D. Patterson, J. Hennessy: Computer Organization & Design - The Hardware/Software Interface, Morgan Kaufmann Publishers, 5. Auflage, 2013
- R. Drechsler, A. Fink, J. Stoppe: Computer – Wie funktionieren Smartphone, Tablet & Co.?, Springer, 2017

Form der Prüfung:

KP, SL:1, PL:1, Portfolio, Klausur, Fachgespräch

Arbeitsaufwand	Präsenz	84 h
	Übungsbetrieb/Prüfungsvorbereitung	186 h
	Summe	270 h

Lehrende:
Prof. Dr. R. Drechsler

Verantwortlich:
Prof. Dr. R. Drechsler

Technische Informatik 2 <i>Technical Computer Science 2</i>							Modulkennziffer: IBGP-T12		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Praktische und Technische Informatik									
Anzahl der SWS	V 4	UE 2	K 0	S 0	Prak. 0	Proj. 0	Σ 6	Kreditpunkte: 9	Turnus angeboten in jedem WiSe
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 3. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> • In der Terminologie der Betriebssysteme und nebenläufigen Systeme kommunizieren können. • Abstraktionshierarchien (Speicherverwaltung, Dateisystem) in Bezug auf ihre Auswirkung auf die Systemleistung einschätzen können. • Lösungsvarianten für Systemsoftwarekomponenten und den Umgang mit Nebenläufigkeit bewerten können. • Schutzmechanismen in Bezug auf Anwendungssicherheitsziele anwenden können. • Selbständiges Entwickeln von einfachen Systemkomponenten in C++ für Unix. • Die globalen Strategien auf einfache vorgegebene Einzelsituationen übertragen können. • In Gruppen Probleme analysieren, gemeinsam Lösungsstrategien entwickeln und präsentieren können. 									
Inhalte: . I. Grundlagen der Betriebssysteme <ul style="list-style-type: none"> • Betriebssysteme: Aufgaben, Rechnerbetriebsformen und Elemente von Betriebssystemen, Anmerkungen zur Geschichte und Überblick über die Entwicklung der Betriebssysteme • Prozessverwaltung: Einfache Prozesse, Prozesseigenschaften, Unterbrechungen, Systemaufrufe, Ausnahmen, Echtzeitbetrieb • Speicherverwaltung: Ein-/Auslagerungsverfahren • Dateisystem: Namen, Baumstruktur; Zugriffsoperationen; Abbildung auf reale Geräte; Ein/Ausgabe; Sicherheit (Schutzmechanismen, Zugriffsrechte) • Befehlsinterpreter II. Nebenläufigkeit <ul style="list-style-type: none"> • Synchronisation: Semaphore, (bedingte) kritische Abschnitte, Ereignisse, Monitore, synchroner/asynchroner Nachrichtenaustausch, "Rendezvous", Kanäle, verteilte Systeme mit Prozedurfernaufrufen • Verklemmungen, Lebendigkeit, Fairness; Korrektheit • Formale Beschreibung nebenläufiger Systeme, z.B. mit Petri-Netzen (Überblick) • Spezielle nebenläufige Systeme: Speisende Philosophen, Erzeuger/Verbraucher, Leser/Schreiber usw. • Grundlagen der Rechnernetze, Client/Server-Architekturen, lokale und globale Netze (Überblick, Ethernet, IP, TCP, HTTP), Sicherheit (Grundlagen der Kryptographie) Lehrveranstaltung(en): <ul style="list-style-type: none"> • 03-IBGP-T12 Technische Informatik 2: Betriebssysteme und Nebenläufigkeit 									
Unterlagen (Skripte, Literatur, Programme usw.): <ul style="list-style-type: none"> • Andrew S. Tanenbaum: Modern Operating Systems, 4th Edition, Pearson Studium, 2016 (bzw. die deutsche Übersetzung: Moderne Betriebssysteme, 4. Auflage, Pearson Studium, 2016) 									

Form der Prüfung:
KP, PL1:40%, PL2:60%, Portfolio, Fachgespräch

Arbeitsaufwand	Präsenz	84 h
	Bearbeitung von Übungsaufgaben/Prüfungsvorbereitung	186 h
	Summe	270 h
Lehrende: Prof. Dr. Ute Bormann		Verantwortlich: Prof. Dr. Ute Bormann

Datenbankgrundlagen & Modellierung <i>Foundations of Data Bases & Modeling</i>							Modulkennziffer: IBGP-DBM		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Praktische und Technische Informatik									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus angeboten in jedem SoSe
	2	2	0	0	0	0	4		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 2. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> • Mit relationalen Datenbanken umgehen. Insbesondere elementare relationale Datenbankschemata und Datenmanipulationsanweisungen verstehen, formulieren und verwenden. • Den Aufbau von Datenbankabfragen kennen und häufig auftretende Anfragen selbstständig formulieren. • UML Diagramme erstellen für statische Aspekte (Klassendiagramme) als auch für dynamische Aspekte (Aktivitäts-, Zustands- und Sequenzdiagramme) • UML-Modelle (mit Klassen, Assoziationen, elementaren Attributtypen, gängigen Multiplizitäten und üblichen Vererbungsstrukturen) in relationale Datenbankschemata transformieren. Insbesondere Schlüssel- und Fremdschlüsselbeziehungen erkennen und nutzen. • Grundideen und Begriffe des relationalen Entwurfs verstehen (Abhängigkeiten zwischen Attributen, Schlüssel, Schlüsselkandidaten, Normalformen, Gütekriterien). 									
Inhalte: <ul style="list-style-type: none"> • Konzepte der Structured Query Language (SQL) • Schemadefinition • Datendefinition • Datenbankabfragen • UML Modellierung • Relationaler Datenbankentwurf Lehrveranstaltung(en): <ul style="list-style-type: none"> • 03-IBGP-DBM Datenbankgrundlagen und Modellierung 									
Unterlagen (Skripte, Literatur, Programme usw.): <ul style="list-style-type: none"> • Alfons Kemper, André Eickler. Datenbanksysteme: Eine Einführung. De Gruyter Oldenbourg; 10. Auflage (25. September 2015) 									
Form der Prüfung: KP; PL1: xx%, PL2: xx%; Portfolio, Klausur									
Arbeitsaufwand	Präsenz		56 h						
	Übungsbetrieb/Prüfungsvorbereitung		124 h						
	Summe		180 h						
Lehrende: Prof. Dr. S. Maneth							Verantwortlich: Prof. Dr. S. Maneth		

Software-Projekt <i>Software Project</i>							Modulkennziffer: IBGP-SWP		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Praktische und Technische Informatik									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus angeboten in jedem WiSe
	0	0	0	0	0	2	2		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 3. Semester									
Sprache: Deutsch									
<p>Ziele:</p> <ul style="list-style-type: none"> • Das angestrebte Ergebnis des Moduls insgesamt ist es, dass die Studierenden die methodischen und praktischen Fähigkeiten erwerben, in einer Gruppe eine Software-Lösung für ein vorgegebenes nicht-triviales Problem zu finden, zu realisieren und zu dokumentieren. • Die zu erwerbenden fachlichen Kompetenzen umfassen mehrere wichtige Aktivitäten in der Softwareentwicklung von der Anforderungsanalyse über den Architekturentwurf bis hin zur Implementierung inklusive systematischen Tests. • Die zu erwerbenden sozialen Kompetenzen betreffen das Projektmanagement in einem Software-Projekt sowie die Gruppenarbeit über einen längeren Zeitraum und die hierfür notwendige Selbstkompetenz (Zeitmanagement, Übernahme von Verantwortung und mehr). 									
<p>Inhalte: Inhaltlich ist das Modul auf eine umfangreichere Aufgabenstellung in größeren Gruppen und für einen längeren Zeitraum ausgelegt. Hierbei gehen die Studierenden mit gelegentlicher Unterstützung der Tutor*innen größtenteils sehr selbstständig und eigenverantwortlich vor. Für eine größere Aufgabenstellung werden über die Dauer eines Semesters mehrere wichtigen Phasen der Software-Entwicklung durchlaufen. Dazu gehören die Anforderungsanalyse, der Architekturentwurf, die Implementierung und das Testen.</p> <p>In größeren Gruppen werden Studierende ein vorgegebenes Problem, das auch die Modellierung von Daten und die Verwendung einer Datenbank umfasst, bearbeiten.</p> <p>Die folgenden, für ein solches Projekt notwendigen Themen der Softwaretechnik werden in Form von Flipped Classroom erarbeitet. Die Studierenden beschäftigen sich im Selbststudium mit vorgegebenen Materialien und besprechen und vertiefen diese in den wöchentlich stattfindenden Übungen und schließlich in der gemeinsamen Projektarbeit.</p> <p>Software-Entwicklungsprozesse</p> <ul style="list-style-type: none"> • Wasserfall-Modell • V-Modell nach B. Boehm <p>Projektplanung</p> <ul style="list-style-type: none"> • Grundbegriffe der Projektplanung • Vorgehen bei der Planung • Inhalt des Projektplans • Gantt-Diagramme und kritischer Pfad • Projektrisiken <p>Anforderungsanalyse</p> <ul style="list-style-type: none"> • Probleme bei der Anforderungsanalyse • Schritte der Anforderungsanalyse • Schritte der Ist-Analyse 									

Inhalte 2: .

- Erhebungstechniken bei der Ist-Analyse (Fragebögen, Interview im Kontext) und Soll-Analyse (Varianten des Prototypings)
- Aufbau und Inhalt der Anforderungsspezifikation
- Produktqualitäten
- Bedeutung und angestrebte Eigenschaften der Anforderungsspezifikation
- Regeln für die Anforderungsspezifikation

Software-Architektur

- Was ist Software-Architektur?
- Sichten (Views) und Blickwinkel (Viewpoints) der Software-Architektur
- Entwurf einer Software-Architektur
- Architekturstile
- Entwurfsmuster
- Modularisierung, Separation of Concern, Abstraktion, Information Hiding

Dokumentation

- interne Software-Dokumentation
- Benutzungshandbücher und Online-Hilfen

Test

- Möglichkeiten und Grenzen des Testens
- Testarten (Komponenten-/Integrations-/Systemtests)
- Test-Varianten: Black-Box, White-Box-Testen
- Testabdeckungsmaße
- Testvorbereitung, -durchführung und -protokollierung

Lehrveranstaltung(en):

- 03-IBGP-SWP Software-Projekt

Unterlagen (Skripte, Literatur, Programme usw.):

- R. Pressman: Software Engineering - A Practitioner's Approach. 6. Auflage, McGraw-Hill, 2004.
- I. Sommerville: Software Engineering. 8. Auflage, Addison-Wesley, 2006.
- W. Zuser, T. Grechenig, M. Köhle: Software Engineering mit UML und dem Unified Process. 2. Auflage, Pearson Studium, 2004.
- Jochen Ludewig, Horst Lichter: Software Engineering - Grundlagen, Menschen, Prozesse, Techniken. dpunkt.verlag, 2006.
- Helmut Balzert: Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering. 3. Auflage, Spektrum Akademischer Verlag, 2009.
- Helmut Balzert: Lehrbuch der Softwaretechnik: Softwaremanagement. 2. Auflage, Spektrum Akademischer Verlag, 2008.
- Chris Rupp: Requirements-Engineering und -Management. 5. Auflage, Hanser Verlag, 2009.
- Klaus Pohl, Chris Rupp: Basiswissen Requirements Engineering. dpunkt.Verlag, 2009.
- Klaus Pohl: Requirements Engineering - Grundlagen, Prinzipien, Techniken. 2. Auflage, dpunkt.Verlag, 2008.
- B. Brügge, A. H. Dutoit: Objektorientierte Softwaretechnik mit UML, Entwurfsmustern und Java. Pearson Studium, 2004.
- Chris Rupp, Stefan Queins, Barbara Zengler: UML 2 glasklar. 3. Auflage, Hanser Verlag, 2007.

Form der Prüfung:

MP, Portfolio (Projektarbeit)

Arbeitsaufwand	Präsenz	28 h
	Übungsbetrieb/Prüfungsvorbereitung	152 h
	Summe	180 h

Lehrende:
Dr. K. Hölscher, Dr. Shi Hui, Amadou

Verantwortlich:
Dr. K. Hölscher

Fachinformatik <i>Application of Computer Science</i>							Modulkennziffer: IBGA-FI		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Angewandte Informatik									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus angeboten in jedem WiSe
	2	2	0	0	0	0	4		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 1. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> • Ein spezifisches Anwendungsfeld der Informatik einordnen können. • Methoden dieses Anwendungsfeldes verstehen und anwenden können. • Anhand exemplarischer Fallbeispiele Gestaltungsoptionen diskutieren und erproben (oder simulieren) können. • In fachübergreifenden Zusammenhängen arbeiten können. • Grundlegende Wechselwirkungen von gesellschaftlichen Rahmenbedingungen und informatischen Umsetzungen erkennen und berücksichtigen können. • In Gruppen Probleme analysieren und gemeinsam Lösungsstrategien entwickeln und präsentieren können. Die inhaltlichen Ziele sind abhängig von der gewählten Lehrveranstaltung.									
Inhalte: Die konkreten Inhalte sind abhängig von der gewählten Lehrveranstaltung. Lehrveranstaltung(en): Auswahl von einer der in diesem Modul angebotenen Lehrveranstaltungen. Derzeit im Angebot sind: <ul style="list-style-type: none"> • 03-B-MI-1.1 Grundlagen der Medieninformatik 1 • 03-IBGA-FI-DG Digitale Gesellschaft • 03-IBGA-FI-ROB Robot Design Lab Angebot wird abhängig von verfügbaren Personalkapazitäten fortgeschrieben. Studierende können auf Antrag an den Prüfungsausschuss stattdessen auch ein einführendes Modul aus einem anderen Studiengang wählen (Nebenfach).									
Unterlagen (Skripte, Literatur, Programme usw.): Abhängig von der gewählten Lehrveranstaltung									
Form der Prüfung: KP; PL1: xx%; PL2: xx%; Portfolio, Fachgespräch, Klausur									
Arbeitsaufwand	Präsenz		56 h						
	Übungsaufgaben/Prüfungsvorbereitung		124 h						
	Summe		180 h						
Lehrende: Verschiedene Dozent/innen							Verantwortlich: Prof. Dr. A.Breiter		

Informatik und Gesellschaft <i>Computer and Society</i>							Modulkennziffer: IBGA-IUG		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Angewandte Informatik									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 3	Turnus angeboten in jedem Semester
	0	0	0	2	0	0	2		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 2. oder 3. Fachsemester									
Sprache: Deutsch									
Kommentar: Aufgrund des Seminarcharakters begrenzte Teilnehmerzahl. Es wird sichergestellt, dass im Laufe eines Studienjahres genügend Plätze für alle Studierenden des Jahrgangs zur Verfügung stehen. Die Platzverteilung in der gemeinsamen Vorbesprechung.									
<p>Ziele: Inhaltlich:</p> <ul style="list-style-type: none"> • Informatik als über rein technische Aspekte hinausreichende Wissenschaft der Gestaltung soziotechnischer Systeme erkennen und diskutieren können. • Gesellschaftliche Wirkungen von Informations- und Kommunikationstechnologien in verschiedenen Bereichen identifizieren und hinterfragen können. • Divergierende Interessen sowie Gestaltungsoptionen beim Einsatz von Informations- und Kommunikationstechniken reflektieren können. • Individuelle und gesellschaftliche Wirkungen des Informationstechnikeinsatzes exemplarisch analysieren, darstellen und bewerten können. • Eigene Positionen zu gesellschaftlichen und ethischen Fragen der Informatik entwickeln und reflektieren. • Informationen und Positionen aus unterschiedlichen Quellen gegenüberstellen können. • Einfache sozialwissenschaftliche Forschungsmethoden auf Gegenstände von Informatik und Gesellschaft anwenden können. <p>General-Studies-Anteile:</p> <ul style="list-style-type: none"> • Erweiterte Recherchemethoden (Bibliothek, Fachdatenbanken und andere Quellen) anwenden können. • Verschiedene Präsentationsformen anwenden und reflektieren können. • Fundiert argumentieren und konstruktiv diskutieren können. • Fachfremde Konzepte und Methoden anhand von Beispielen verstehen können. • In Gruppen Probleme analysieren und gemeinsam Lösungsstrategien entwickeln und präsentieren können. 									

Inhalte: .

1 Sichtweisen der Informatik: Wissenschaftstheoretische und ethische Aspekte der Informatik; Entwicklung der Disziplin; Verantwortung der Informatiker und Informatikerinnen; Computer als Werkzeug und Medium; Formalisierung und Modellbildung; ...;

2 Arbeit: Konzepte der Automation und ökonomische Rahmenbedingungen der Automatisierung: Betriebliche Wirkungen des Rechneinsatzes; gesamtgesellschaftliche und gesamtwirtschaftliche Wirkungen; Arbeitsmarktentwicklung unter dem Einfluss des Informationstechnikeinsatzes, Ansätze zur Gestaltung computergestützter Arbeitssysteme; neue Formen der Arbeit; Mitbestimmung; ...;

3 Sozialisation, Bildung und Persönlichkeit: Digitale Medien in der Bildung; Kommunikation und soziale Netzwerke; Digitale Medien und Identität; Genderaspekte; Technikfaszination und -akzeptanz; ...;

4 Informatisierung des Alltags: Digitale Medien und Kultur; Computerspiele; Informationstechnik und Behinderung; Konsum und Kommerz; mobile und ubiquitous computing; Service-Robotik; Beschleunigung der Gesellschaft; ...;

5 Spezifische Einsatzfelder von Informations- und Kommunikationstechnik, z.B.:

- Innere und äußere Sicherheit: Polizei, Militär, Überwachung, ...;
- Umwelt: Umweltfolgen der Informationstechnik, Beitrag der Informatik zum Umweltschutz, ...;
- Gesundheitswesen: Informatik im Krankenhaus, Informatik in der Arztpraxis, ...;
- Politik: Partizipation, Internet und Demokratie, Online-Wahlen, ...;
- Globalisierung: Informatik und „3. Welt“; ...; u.a.m.

6 Datenschutz: Abgrenzung Datenschutz und Datensicherheit; verfassungsrechtliche und gesetzliche Grundlagen, Prinzipien und Institutionen des Datenschutzes; rechtliche, technische und organisatorische Maßnahmen des Datenschutzes; Datenschutz durch Technikgestaltung; Datenschutz im Betrieb; Datenschutz im Internet; ...;

7 Rechtliche Fragen von IT-Entwicklung und –Einsatz: Multimedia-Gesetze; Lizenzen / Open Source; Softwarepatente; Urheberrechte; Kryptographie-Debatte; Computerkriminalität; ...;

Lernmethoden: Während die meisten Pflichtmodule in der Studienanfängersphase in Form von Vorlesungen mit Übungen durchgeführt werden, wurde für „Informatik und Gesellschaft“ bewusst die Seminarform gewählt, da diese besonders geeignet ist für die kontroverse Diskussion und Erörterung von Positionen, Bewertungen und Werten.

1 Referat (bis zu 3 Personen)

- mündlicher Vortrag zu einem ausgewählten Thema (ca. 30-45 Minuten) und anschließende Diskussion;
- schriftliche Ausarbeitung der Präsentation unter Berücksichtigung von in der Diskussion ergänzend eingebrachten relevanten Informationen;

2 Vorbereiteter Diskussionsbeitrag zu einer anderen Präsentation;

Lehrveranstaltung(en):

- 03-IBGA-IUG Informatik und Gesellschaft

Unterlagen (Skripte, Literatur, Programme usw.): Zu Beginn des Seminars erfolgt im Rahmen einer Seminarsitzung eine unterstützte, themenspezifische Literaturrecherche in der Bibliothek. Weitere Hintergrund- bzw. Überblicksliteratur:

Zeitschriften u.a.:

“Flif-Kommunikation” - SuUB: z inf 034 j/896

“Datenschutz-Nachrichten: DANA” - SuUB: z inf 054 j/350

“Datenschutz und Datensicherheit, Recht und Sicherheit in Informationsverarbeitung und Kommunikation: DuD” - SuUB: z jur 018.5/500 (Standort: Juridicum GW1) <http://www.springerlink.com/content/1862-2607/> (Zugang im Campus-Netz)

“Datenschutz-Berater: DSB” - http://www.wiso-net.de/webcgi?START=DC0&IV_DBN=DSB (Zugang im Campus-Netz)

“Computer und Arbeit: CuA”

“Vorgänge: Zeitschrift für Bürgerrechte und Gesellschaftspolitik” - SuUB: z sow 006/545

“Bürgerrechte & Polizei: CILIP” - SuUB: z jur 240/200 (Standort: Juridicum GW1)

“Zeitschrift für Urheber- und Medienrecht: ZUM” – SuUB: z tea 930 ja/213

Bücher:

Weber-Wulff, D.; Class, Ch.; Coy, W.; Kurz, C.; Zellhöfer, D. (2009): *Gewissensbisse : ethische Probleme der Informatik. Biometrie - Datenschutz - geistiges Eigentum.* Bielefeld: transcript. - SuUB: a inf 036/354 (und andere Exemplare)

Adams, A.A.; McCrindle, R.J. (2008): *Pandora's box: social and professional issues of the information age.* Chicester: Wiley. - SuUB: a inf 036 e/321

Baase, S. (2008): *A Gift of Fire: Social, Legal, and Ethical Issues for Computing and the Internet (3rd Edition).* Prentice Hall. - SuUB: /bestellt/

Barger, R.N. (2008): *Computer ethics: a case-based approach.* Cambridge: Cambridge Univ. Press. - SuUB: a inf 036 e/001

Rolf, A. (2008): *Mikropolis 2010: Menschen, Computer, Internet in der globalen Gesellschaft.* Marburg: Metropolis. - SuUB: a inf 032/793

Roßnagel, A.; Winand, U.; Sommerlatte, T. (Hrsg.) (2008): *Digitale Visionen: Zur Gestaltung allgegenwärtiger Informationstechnologien.* Berlin: Springer. - <http://dx.doi.org/10.1007/978-3-540-77022-0> (Zugang im Campus-Netz)

Kizza, J.M. (2003): *Ethical and social issues in the information age.* New York: Springer. - SuUB: a soz 312.7 ea/212(2)

Fuchs, Ch.; Hofkirchner, W. (2003): *Studienbuch Informatik und Gesellschaft.* Norderstedt: Books on Demand. - SuUB: TB BHV com 10/60 (Standort: Bremerhaven)

Spinello, R.A. (2002): *Case Studies in Information Technology Ethics (2nd Edition).* Prentice Hall. - SuUB: /bestellt/

Tübinger Studentexte Informatik und Gesellschaft (1999) (9 Hefte von verschiedenen AutorInnen). - SuUB: 01.K.6857

Friedrich, J.; Herrmann, T.; Peschek, M.; Rolf, A. (Hrsg.) (1995): *Informatik und Gesellschaft.* Heidelberg: Spektrum. - SuUB: a inf 030 e/705 (und weitere Exemplare)

Steinmüller, W. (1993): *Informationstechnologie und Gesellschaft.* Darmstadt: Wiss. Buchges. - SuUB: a inf 800 e/040 (und weitere Exemplare)

Form der Prüfung:

KP; PL1: xx%, PL2: xx%;Referat+Ausarbeitung, Fachgespräch, Bonusprüfung

Arbeitsaufwand	Präsenz	28 h
	Vortrag vorbereiten/Ausarbeitung schreiben	62 h
	Summe	90 h
Lehrende: R.E. Streibl		Verantwortlich: R.E. Streibl

Angewandte Informatik <i>Applied Computer Science</i>							Modulkennziffer: IBGA-AI		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Grundlagen Angewandte Informatik									
Anzahl der SWS	V 2	UE 2	K 0	S 0	Prak. 0	Proj. 0	Σ 4	Kreditpunkte: 6	Turnus angeboten in jedem SoSe
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 4. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> Studierende kennen und verstehen theoretische Grundlagen von sozio-technischen Systemen in ausgewählten Anwendungskontexten Studierende können die Rolle von Informationssystemen in unterschiedlichen, organisationalen und gesellschaftlichen Anwendungskontexten erläutern und mit Hilfe ausgewählter Theoriekonzepte kritisch analysieren und reflektieren Studierende können ethische und soziale Fragestellungen in Bezug auf Entwicklung und Nutzung von Informationssystemen erläutern, analysieren und bewerten Studierende kennen Methoden zur Analyse und Modellierung von Informationssystemen; sie können sie zur Erhebung und Analyse von IT-gestützten Geschäftsprozessen anwenden und bewerten Studierende lernen verschiedene Rechtsbereiche kennen, die Einfluss auf das Handeln von Informatiker*innen haben 									
Inhalte: <ul style="list-style-type: none"> Angewandte Informatik und die Verortung in Wirtschaftsinformatik (Information Systems Research), Mensch-Maschine-Interaktion (HCI) und Science and Technology Studies (STS) Grundlagen sozio-technischer Systeme Informatiksysteme in spezifischen Anwendungskontexten (Wirtschaft, Verwaltung, Bildung) Ethische, rechtliche und soziale Aspekte von Informatiksystemen (ELSA) Grundlagen des Informations- und Wissensmanagements Informationssysteme und Methoden der Geschäftsprozessmodellierung anhand von Fallbeispielen in Anwendungsbereichen 									
Lehrveranstaltung(en): <ul style="list-style-type: none"> 03-IBGA-AI Grundlagen der angewandten Informatik 									
Unterlagen (Skripte, Literatur, Programme usw.): <ul style="list-style-type: none"> Ausgewählte Aufsätze aus den Bereichen Information Systems, HCI und STS. Barry, A. (2001). Political machines: Governing a technological society. London, New York: Athlone Press. Bijker, W. E., Hughes, T. P., & Pinch, T. J. (Eds.). (1987). The social construction of technological systems: New directions in the sociology and history of technology. Cambridge, Mass: MIT Press. Suchman, L. (2007). Human-machine reconfigurations: Plans and situated actions. Cambridge: Cambridge Univ. Press Winograd, T., & Flores, F. (1986). Understanding Computers and Cognition. A New Foundation for Design. Norwood, NJ: Ablex. 									
Form der Prüfung: KP, PL1: xx%, PL2: xx%, Portfolio, Fachgespräch									

Arbeitsaufwand	Präsenz	56 h
	Virtuelle Vorlesung (inkl. Nachbereitung)	24 h
	Übungsbetrieb/Prüfungsvorbereitung	100 h
	Summe	180 h
Lehrende: Prof. Dr. A. Breiter, N.N.		Verantwortlich: Prof. Dr. A. Breiter

Aufbau Theoretische Informatik <i>Theoretical Computer Science (Intermediate Level)</i>							Modulkennziffer: IBAT			
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>										
Studienabschnitt: Aufbau										
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus I.d.R. Angebote in jedem Semester	
	0	0	4	0	0	0	4			
Formale Voraussetzungen: -										
Inhaltliche Voraussetzungen: Kompetenzen aus bestimmten Grundlagenmodulen abhängig von der gewählten Lehrveranstaltung.										
Vorgesehenes Semester: ab 5. Semester										
Sprache: Deutsch/Englisch										
Ziele: Die Studierenden erwerben aufbauend auf den in den Grundlagenmodulen erworbenen Kompetenzen ein grundlegendes Verständnis eines Teilgebietes der Theoretischen Informatik. Sie kennen damit exemplarisch den formalen und mathematisch präzisen Zugang zu Themen der Informatik. Zudem sind sie vertraut mit den zentralen formalen Begriffen des betreffenden Teilgebiets und dessen essentiellen Theoremen, Beweis- und Analysemethoden. Die Studierenden können die wichtigsten Resultate und Konstruktionen des Gebietes sowie grundlegende Beweismethoden selbständig anwenden. Die erworbenen Kompetenzen sind inhaltliche Voraussetzung für vertiefende Lehrangebote im jeweiligen Teilgebiet. Die konkreten Kompetenzen sind abhängig von der gewählten Lehrveranstaltung.										
Inhalte: Die konkreten Inhalte sind abhängig von der gewählten Lehrveranstaltung. Lehrveranstaltung(en): Auswahl von einer der in diesem Modul angebotenen Lehrveranstaltungen. Derzeit im Angebot sind: <ul style="list-style-type: none"> • 03-IBAT-AAG Algorithmen auf Graphen • 03-IBAT-LO Logik • 03-IBAT-PN Petri-Netze • 03-IBAT-KS Korrekte Software • 03-IBAT-OR Operations Research [Angebot wird abhängig von verfügbaren Personalkapazitäten fortgeschrieben]										
Unterlagen (Skripte, Literatur, Programme usw.): Abhängig von der gewählten Lehrveranstaltung.										
Form der Prüfung: MP; Portfolio, Fachgespräch, mündliche Prüfung, Klausur, Hausarbeit, Referat+Ausarbeitung, ggf. Bonusprüfung										
Arbeitsaufwand	Präsenz		56 h		Bearbeitung von Übungsaufgaben/Prüfungsvorbereitung		124 h		Summe	180 h
Lehrende: Verschiedene Dozent/innen					Verantwortlich: Prof. Dr. Carsten Lutz					

Aufbau Praktische Informatik <i>Practical Computer Science (Intermediate Level)</i>							Modulkennziffer: IBAP			
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>										
Studienabschnitt: Aufbau										
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus Angebote in jedem Semester	
	0	0	4	0	0	0	4			
Formale Voraussetzungen: -										
Inhaltliche Voraussetzungen: Kompetenzen aus bestimmten Grundlagenmodulen abhängig von der gewählten Lehrveranstaltung.										
Vorgesehenes Semester: ab 4. Semester										
Sprache: Deutsch/Englisch										
Ziele: Die Studierenden erwerben aufbauend auf den in den Grundlagenmodulen erworbenen Kompetenzen ein grundlegendes Verständnis eines Teilgebietes der Praktischen Informatik. Sie kennen (Architektur)Konzepte, Modellierungsverfahren und/oder Algorithmen und damit die in der jeweiligen Domäne spezifischen Methoden zur Entwicklung von Software und können diese auf einfache Aufgabenstellungen im jeweiligen Teilgebiet anwenden. Die erworbenen Kompetenzen sind inhaltliche Voraussetzung für vertiefende Veranstaltungen im jeweiligen Teilgebiet. Die konkreten Kompetenzen sind abhängig von der gewählten Lehrveranstaltung.										
Inhalte: Die konkreten Inhalte sind abhängig von der gewählten Lehrveranstaltung. Lehrveranstaltung(en): Auswahl von einer der in diesem Modul angebotenen Lehrveranstaltungen. Derzeit im Angebot sind:										
<ul style="list-style-type: none"> • 03-IBAP-RA Rechnerarchitektur und eingebettete Systeme • 03-IBAP-BS Betriebssysteme • 03-IBAP-DBS Datenbanksysteme • 03-IBAP-RN Rechnernetze • 03-IBAP-ÜB Übersetzerbau • 03-IBAP-SWT Softwaretechnik • 03-IBAP-ISEC Informationssicherheit • 03-IBAP-CG Computergraphik • 03-IBAP-SDV Sensordatenverarbeitung • 03-IBAP-KI Grundlagen der Künstlichen Intelligenz • 03-IBAP-ML Grundlagen des Maschinellen Lernens • 03-IBAP-CS Cognitive Systems • 03-IBAP-MRCA Modern Robot Control Architectures 										
[Angebot wird abhängig von verfügbaren Personalkapazitäten fortgeschrieben]										
Unterlagen (Skripte, Literatur, Programme usw.): Abhängig von der gewählten Lehrveranstaltung										
Form der Prüfung: MP; Portfolio, Fachgespräch, mündliche Prüfung, Klausur, Hausarbeit, Referat+Ausarbeitung, ggf. Bonusprüfung										
Arbeitsaufwand	Präsenz		56 h		Bearbeitung von Übungsaufgaben/Prüfungsvorbereitung		124 h		Summe	180 h
Lehrende: Verschiedene Dozent/innen					Verantwortlich: Prof. Dr. U. Bormann					

Aufbau Informatik <i>Computer Science (Intermediate Level)</i>							Modulkennziffer: IBA		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Aufbau									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 12	Turnus Angebote in jedem Semester
	0	0	8	0	0	0	8		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: Kompetenzen aus bestimmten Grundlagenmodulen abhängig von den gewählten Lehrveranstaltungen.									
Vorgesehenes Semester: ab 4. Semester									
Sprache: Deutsch/Englisch									
Ziele: Die Studierenden erwerben aufbauend auf den in den Grundlagenmodulen erworbenen Kompetenzen ein grundlegendes Verständnis von zwei weiteren Teilgebieten der Informatik. Dabei kann es sich um Teilgebiete der Theoretischen Informatik, der Praktischen Informatik und/oder der Angewandten Informatik handeln.									
Inhalte: Abhängig von den beiden gewählten Lehrveranstaltungen.									
Lehrveranstaltungen:									
Auswahl von einer der in diesem Modul angebotenen Lehrveranstaltungen. Derzeit im Angebot sind:									
Aufbau Theoretische Informatik:									
<ul style="list-style-type: none"> ● 03-IBAT-AAG Algorithmen auf Graphen ● 03-IBAT-LO Logik ● 03-IBAT-PN Petri-Netze ● 03-IBAT-KS Korrekte Software ● 03-IBAT-OR Operations Research 									
Aufbau Praktische Informatik:									
<ul style="list-style-type: none"> ● 03-IBAP-RA Rechnerarchitektur und eingebettete Systeme ● 03-IBAP-BS Betriebssysteme ● 03-IBAP-DBS Datenbanksysteme ● 03-IBAP-RN Rechnernetze ● 03-IBAP-ÜB Übersetzerbau ● 03-IBAP-SWT Softwaretechnik ● 03-IBAP-ISEC Informationssicherheit ● 03-IBAP-CG Computergraphik ● 03-IBAP-SDV Sensordatenverarbeitung ● 03-IBAP-KI Grundlagen der Künstlichen Intelligenz ● 03-IBAP-ML Grundlagen des Maschinellen Lernens ● 03-IBAP-CS Cognitive Systems ● 03-IBAP-MRCA Modern Robot Control Architectures 									

Inhalte 2: .

Aufbau Angewandte Informatik

- 03-IBAA-MTI Mensch-Technik-Interaktion
- 03-IBAA-ITM Informationstechnikmanagement
- 03-IBAA-DS Datenschutz
- 03-IBAA-ECA E-Commerce Anwendungen
- 03-IBAA-EM Empirische Methoden für Informatik/Digitale Medien
- 03-IBAA-BUB Biosignale und Benutzerschnittstellen
- 03-B-MI-1.2 Grundlagen der Medieninformatik 2

[Angebot wird abhängig von verfügbaren Personalkapazitäten fortgeschrieben. Die in den Modulen IMAT und IMAP ausgewählten Lehrangebote stehen hier nicht erneut zur Verfügung.]

Unterlagen (Skripte, Literatur, Programme usw.): Abhängig von den gewählten Lehrveranstaltungen

Form der Prüfung:

TP; PL1 50%, PL2 50%; Portfolio, Fachgespräch, mündliche Prüfung, Klausur, Hausarbeit, Referat+Ausarbeitung, ggf. Bonusprüfung

Arbeitsaufwand	Präsenz	112 h
	Bearbeitung von Übungsaufgaben/Prüfungsvorbereitung	248 h
	Summe	360 h
Lehrende: Verschiedene Dozent/innen		Verantwortlich: Prof. Dr. U. Bormann

Vertiefung Theoretische Informatik <i>Advanced Theoretical Computer Science</i>							Modulkennziffer: IBVT		
Bachelor Pflicht <input type="checkbox"/> Wahlpflicht <input checked="" type="checkbox"/>									
Studienabschnitt: Vertiefung									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus Angebote in jedem Semester
	0	0	4	0	0	0	4		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: Kompetenzen aus bestimmten Grundlagen- oder Aufbaumodulen abhängig von der gewählten Lehrveranstaltung.									
Vorgesehenes Semester: ab 5. Semester									
Sprache: Deutsch/Englisch									
Kommentar: Dieses Modul ist eine der drei Wahlpflicht-Alternativen von „Vertiefung Informatik“									
Ziele: Die Studierenden erwerben weitere Kompetenzen aus dem Bereich der Theoretischen Informatik. Dabei kann es sich sowohl um Einblicke in ein weiteres Themenfeld der Theoretischen Informatik als auch um vertiefte Kompetenzen handeln. Die inhaltlichen Lernziele sind abhängig von der gewählten Lehrveranstaltung.									
Inhalte: Die konkreten Inhalte sind abhängig von der gewählten Lehrveranstaltung. Lehrveranstaltung(en): Auswahl von einer der in diesem Modul angebotenen Lehrveranstaltungen: <ul style="list-style-type: none"> • Vertiefungs-Lehrveranstaltungen der Theoretischen Informatik (03-IBVT-xx) • Aufbau-Lehrveranstaltungen der Theoretischen Informatik (03-IBAT-xx), sofern nicht bereits in anderem Modul belegt. • Bei Vorliegen entsprechender Vorkenntnisse können auch Aufbau- oder Vertiefungs-Lehrveranstaltungen der Theoretischen Informatik aus dem Lehrangebot des Master-Studiengangs Informatik besucht werden (03-IMAT-xx, 03-IMVT-xx) Die konkreten Angebote werden abhängig von verfügbaren Personalkapazitäten fortgeschrieben und können der Vier-Semester-Planung und dem Veranstaltungsverzeichnis entnommen werden.									
Unterlagen (Skripte, Literatur, Programme usw.): Abhängig von der gewählten Lehrveranstaltung.									
Form der Prüfung: MP; Portfolio, Fachgespräch, mündliche Prüfung, Klausur, Hausarbeit, Referat+Ausarbeitung, ggf. Bonusprüfung									
Arbeitsaufwand	Präsenz		56 h						
	Bearbeitung von Übungsaufgaben/Prüfungsvorbereitung		124 h						
	Summe		180 h						
Lehrende: Verschiedene Dozent/innen					Verantwortlich: Prof. Dr. Carsten Lutz				

Vertiefung Praktische Informatik <i>Advanced Practical Computer Science</i>							Modulkennziffer: IBVP		
Bachelor Pflicht <input type="checkbox"/> Wahlpflicht <input checked="" type="checkbox"/>									
Studienabschnitt: Vertiefung									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus Angebote in jedem Semester
	0	0	4	0	0	0	4		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: Kompetenzen aus bestimmten Grundlagen- oder Aufbaumodulen abhängig von der gewählten Lehrveranstaltung.									
Vorgesehenes Semester: ab 5. Semester									
Sprache: Deutsch/Englisch									
Kommentar: Dieses Modul ist eine der drei Wahlpflicht-Alternativen von „Vertiefung Informatik“									
Ziele: Die Studierenden erwerben weitere Kompetenzen aus dem Bereich der Praktischen Informatik. Dabei kann es sich sowohl um Einblicke in ein weiteres Themenfeld der Praktischen Informatik als auch um vertiefte Kompetenzen handeln. Die inhaltlichen Lernziele sind abhängig von der gewählten Lehrveranstaltung.									
Inhalte: Die konkreten Inhalte sind abhängig von der gewählten Lehrveranstaltung. Lehrveranstaltung(en): Auswahl von einer der in diesem Modul angebotenen Lehrveranstaltungen: <ul style="list-style-type: none"> • Vertiefungs-Lehrveranstaltungen der Praktischen Informatik (03-IBVP-xx) • Aufbau-Lehrveranstaltungen der Praktischen Informatik (03-IBAP-xx), sofern nicht bereits in anderem Modul belegt. • Bei Vorliegen entsprechender Vorkenntnisse können auch Aufbau- oder Vertiefungs-Lehrveranstaltungen der Praktischen Informatik aus dem Lehrangebot des Master-Studiengangs Informatik besucht werden (03-IMAP-xx, 03-IMVP-xx) Die konkreten Angebote werden abhängig von verfügbaren Personalkapazitäten fortgeschrieben und können der Vier-Semester-Planung und dem Veranstaltungsverzeichnis entnommen werden.									
Unterlagen (Skripte, Literatur, Programme usw.): Abhängig von der gewählten Lehrveranstaltung.									
Form der Prüfung: MP; Portfolio, Fachgespräch, mündliche Prüfung, Klausur, Hausarbeit, Referat+Ausarbeitung, ggf. Bonusprüfung									
Arbeitsaufwand	Präsenz		56 h						
	Bearbeitung von Übungsaufgaben/Prüfungsvorbereitung		124 h						
	Summe		180 h						
Lehrende: Verschiedene Dozent/innen					Verantwortlich: Prof. Dr. Ute Bormann				

Vertiefung Angewandte Informatik <i>Advanced Applied Computer Science</i>							Modulkennziffer: IBVA		
Bachelor Pflicht <input type="checkbox"/> Wahlpflicht <input checked="" type="checkbox"/>									
Studienabschnitt: Vertiefung									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus Angebote in jedem Semester
	0	0	4	0	0	0	4		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: Kompetenzen aus bestimmten Grundlagen- oder Aufbaumodulen abhängig von der gewählten Lehrveranstaltung.									
Vorgesehenes Semester: ab 5. Semester									
Sprache: Deutsch/Englisch									
Kommentar: Dieses Modul ist eine der drei Wahlpflicht-Alternativen von „Vertiefung Informatik“									
Ziele: Die Studierenden erwerben weitere Kompetenzen aus dem Bereich der Angewandten Informatik. Dabei kann es sich sowohl um Einblicke in ein weiteres Themenfeld der Angewandten Informatik als auch um vertiefte Kompetenzen handeln. Die inhaltlichen Lernziele sind abhängig von der gewählten Lehrveranstaltung.									
Inhalte: Die konkreten Inhalte sind abhängig von der gewählten Lehrveranstaltung. Lehrveranstaltung(en): Auswahl von einer der in diesem Modul angebotenen Lehrveranstaltungen: <ul style="list-style-type: none"> • Vertiefungs-Lehrveranstaltungen der Angewandten Informatik (03-IBVA-xx) • Aufbau-Lehrveranstaltungen der Angewandten Informatik (03-IBAA-xx), sofern nicht bereits in anderem Modul belegt. • Bei Vorliegen entsprechender Vorkenntnisse können auch Aufbau- oder Vertiefungs-Lehrveranstaltungen der Angewandten Informatik aus dem Lehrangebot des Master-Studiengangs Informatik besucht werden (03-IMAA-xx, 03-IMVA-xx) Die konkreten Angebote werden abhängig von verfügbaren Personalkapazitäten fortgeschrieben und können der Vier-Semester-Planung und dem Veranstaltungsverzeichnis entnommen werden.									
Unterlagen (Skripte, Literatur, Programme usw.): Abhängig von der gewählten Lehrveranstaltung.									
Form der Prüfung: MP; Portfolio, Fachgespräch, mündliche Prüfung, Klausur, Hausarbeit, Referat+Ausarbeitung, ggf. Bonusprüfung									
Arbeitsaufwand	Präsenz		56 h						
	Bearbeitung von Übungsaufgaben/Prüfungsvorbereitung		124 h						
	Summe		180 h						
Lehrende: Verschiedene Dozent/innen					Verantwortlich: Prof. Dr. Andreas Breiter				

Bachelorprojekt (Teil 1) <i>Bachelor Project (Part 1)</i>							Modulkennziffer: IBPJ1		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Vertiefung									
Anzahl der SWS	V 0	UE 0	K 0	S 0	Prak. 0	Proj. 4	Σ 4	Kreditpunkte: 12	Turnus Angebote in jedem WiSe
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 5. Semester									
Sprache: Deutsch									
<p>Ziele: Im Projekt wird ein größeres Vorhaben umgesetzt. Außer den für jedes Projekt jeweils spezifischen fachlichen Zielen werden zusätzlich Metaziele verfolgt. Von den hier beschriebenen Zielen ist eine gewisse Bandbreite umzusetzen: Jedes Projekt soll alle Bereiche A, B, C umschließen und daraus jeweils mehrere Ziele verfolgen.</p> <p>A Qualität professioneller Systementwicklung</p> <ol style="list-style-type: none"> 1. Geeignete Methoden für Aufgabenanalyse, Spezifikation und Systementwicklung im Kontext eines größeren Projekts anwenden können 2. Für ein spezifisches informationstechnisches Anwendungsfeld Programmiersprachen und Programmierumgebungen auswählen und benutzen, sowie bestehenden Quellcode lesen und modifizieren können 3. Im Kontext des Projekts ggf. Methoden des Interaction Designs, User Centered Design und Experience Design anwenden, sowie verschiedene Designentwürfe vergleichen und bewerten können 4. Methoden der Evaluation, Testverfahren, Qualitätsmanagement und Dokumentation einsetzen können 5. Ggf. das regulatorische Umfeld (Standards, Zertifizierung, Lizenzierung, Open Source, etc.) erkennen und verstehen <p>B Forschungspraxis und Wissenschaftskultur</p> <ol style="list-style-type: none"> 1. Das projektspezifische Forschungsfeld exemplarisch erfahren und einschlägige Fachliteratur recherchieren und verstehen können 2. Eigene wissenschaftliche Texte schreiben können (Dokumentation, Projektbericht, etc.) 3. Fachliche Netzwerke, Wissenschaftsorganisationen und –kulturen im projektspezifischen Bereich kennen (Foren, Tagungen, Fachgesellschaften, Publikationen, etc.) <p>C „Soft Skills“</p> <ol style="list-style-type: none"> 1. Aufgaben und Methoden des Projektmanagements kennen und im Projektkontext anwenden können (Planung, Zeit- und Arbeitsorganisation, Aufwandsmessung, Business Plan, etc.) 2. Soziale, rechtliche, ökonomische und technische Rahmenbedingungen analysieren und für den Projektkontext bewerten können 3. Dimension der gesellschaftlichen Verantwortung für den Projektkontext analysieren, verstehen, diskutieren und bewerten können (Ambivalenzen, Interessen, ethische Leitlinien, etc.) 4. In der Projektpraxis zu einer vertieften interkulturelle Kompetenz zu kommen 5. Genderaspekte verstehen und erkennen sowie Gleichstellungsorientierung in der Praxis anwenden können 6. Kommunikative Kompetenz (Diskussionsfähigkeit, Moderation, Konfliktmanagement) praktizieren können, dabei die Fähigkeit zur Teamarbeit erwerben, andererseits auch Leitungsaufgaben übernehmen können 7. Präsentationsfähigkeit und Öffentlichkeitsarbeit für universitäre und außeruniversitäre Adressaten beherrschen <p>Insbesondere die Kompetenzen A4, B2 und C7 werden im 2. Teil des Projekts im darauffolgenden Semester noch deutlich vertieft.</p>									

Inhalte: Die fachlichen Inhalte sind projektspezifisch und können daher nicht allgemein beschrieben werden.

Das Projekt wird im darauffolgenden Semester fortgesetzt.

Lehrveranstaltung(en):

- Auswahl aus den im jeweiligen Jahrgang angebotenen konkreten Bachelorprojekten.

Unterlagen (Skripte, Literatur, Programme usw.): Projektspezifisch

Form der Prüfung:

MP, Projektarbeit

Arbeitsaufwand	Präsenz im Projektplenum	80 h
	eigentliche Projektarbei	280 h
	Summe	360 h

Lehrende:

Im Wechsel Angebote aus diversen Arbeitsgruppen

Verantwortlich:

Prof. Dr. U. Bormann

Bachelorprojekt (Teil 2) <i>Bachelor Project (Part 2)</i>							Modulkennziffer: IBPJ2		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Vertiefung									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 3	Turnus Angebote in jedem SoSe
	0	0	0	0	0	2	2		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 6. Semester									
Sprache: Deutsch									
<p>Ziele: Im Projekt wird ein größeres Vorhaben umgesetzt. Außer den für jedes Projekt jeweils spezifischen fachlichen Zielen werden zusätzlich Metaziele verfolgt.</p> <p>Bei diesem Modul handelt es sich um eine Fortsetzung des Moduls „Bachelorprojekt (Teil 1)“. In diesem zweiten Teil des Projekts sollen insbesondere die folgenden Kompetenzen vertieft werden:</p> <ul style="list-style-type: none"> • Methoden der Evaluation, Testverfahren, Qualitätsmanagement und Dokumentation einsetzen können. • Eigene wissenschaftliche Texte schreiben können (Dokumentation, Projektbericht, etc.) • Präsentationsfähigkeit und Öffentlichkeitsarbeit für universitäre und außeruniversitäre Adressaten beherrschen 									
<p>Inhalte: Bei diesem Modul handelt es sich um eine Fortsetzung des Moduls „Bachelorprojekt (Teil 1)“.</p> <p>Die fachlichen Inhalte sind projektspezifisch und können daher nicht allgemein beschrieben werden.</p>									
Unterlagen (Skripte, Literatur, Programme usw.): Projektspezifisch									
Form der Prüfung: MP, Projektarbeit									
Arbeitsaufwand	Präsenz im Projektplenum		16 h		eigentliche Projektarbeit		74 h		
	Summe		90 h						
Lehrende: Im Wechsel Angebote aus diversen Arbeitsgruppen							Verantwortlich: Prof. Dr. U. Bormann		

Bachelorarbeit <i>Bachelor Thesis</i>							Modulkennziffer: IBR		
Bachelor Pflicht <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>									
Studienabschnitt: Bachelorarbeit									
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 12	Turnus Kann jederzeit mit Betreuenden vereinbart werden
	0	0	0	0	0	0	0		
Formale Voraussetzungen: Mathematik 1, Theoretische Informatik 1, Praktische Informatik 1, Praktische Informatik 2, Datenbankgrundlagen und Modellierung, Software-Projekt, sowie mind. 120 CP absolviert									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: 6. Semester									
Sprache: Deutsch/Englisch									
Kommentar: Keine regelmäßigen Präsenzzeiten, daher keine expliziten SWS ausgewiesen. Allerdings wird in der betreuenden Arbeitsgruppe oft ein Graduierten-Seminar zur Präsentation von Zwischenständen der Abschlussarbeit angeboten. Die Teilnahme daran ist dann integraler Bestandteil des Moduls Bachelorarbeit.									
Ziele: Die inhaltlichen Ziele sind abhängig vom gewählten Thema. Metaziele: Die Studierenden verfügen über <ul style="list-style-type: none"> • Methoden, um Aufgaben mit den Mitteln der Informatik zeit- und kostengerecht lösen und insbesondere die eigene Arbeit organisieren zu können. • Grundkenntnisse im Schätzen und Messen von Aufwand und Produktivität • Fähigkeit zur Bearbeitung von Aufgaben in einem gewissen Anwendungsfeld unter gegebenen technischen, ökonomischen und sozialen Randbedingungen mit Mitteln der Informatik, • Fähigkeit zur Entwicklung entsprechender Systeme • Fähigkeit, Anwendungsprobleme im Gesamtzusammenhang zu erkennen, Vertrautheit mit zugehörigen Lösungsmustern • Fähigkeit zum professionellen Erstellen und Testen größerer Softwaresysteme • Fähigkeit, sich in vorhandene Programme einzuarbeiten und vorhandene Programmelemente nutzen zu können. • Fähigkeit zur Erarbeitung von Lösungen (bei begrenzten Ressourcen), die allgemein anerkannten Qualitätsstandards genügen, • Kommunikative Kompetenz, um Ideen und Lösungsvorschläge schriftlich und mündlich überzeugend zu präsentieren, • Fähigkeit zur wissenschaftlichen Arbeit, zum Wissenserwerb sowie Transferkompetenz • Bei einer Gruppenarbeit auch Fähigkeit zur Teamarbeit 									
Inhalte: Die Inhalte sind abhängig vom gewählten Thema.									
Unterlagen (Skripte, Literatur, Programme usw.): Themenspezifisch									
Form der Prüfung: MP, PL1: 67%, PL2: 33%, Thesis, Kolloquium									
Arbeitsaufwand	Bearbeitung der Aufgabenstellung				300 h				
	Vorbereitung und Durchführung des Kolloquiums				60 h				
	Summe				360 h				
Lehrende: Alle selbständig Lehrenden können Bachelorarbeiten betreuen					Verantwortlich: Prof. Dr. U. Bormann				