

Modul 1: Formale Grundlagen der Informatik

Modul-Kennnummer (JOGU-StINe)	Arbeitsaufwand (workload)	Moduldauer (laut Studienverlaufsplan)	Regelsemester (laut Studienverlaufsplan)	Leistungspunkte (LP)
	240 h	1 Semester	1. Semester	8 LP
1.	Lehrveranstaltungen/Lehrformen	Kontaktzeit	Selbststudium	Leistungspunkte
	a) Diskrete Mathematik – Vorlesung (P)	4 SWS/42 h	108 h	5 LP
	b) Übung zur Vorlesung a) (P)	2 SWS/21 h	69 h	3 LP
2.	Gruppengrößen Gemäß aktueller Satzung über die Betreuungsrelationen von Lehrveranstaltungen in Bachelor- und Masterstudiengängen und zur Festsetzung der Normwerte für den Ausbildungsaufwand (Curriculumnormwerte) der Johannes Gutenberg-Universität Mainz (http://www.uni-mainz.de/studlehr/ordnungen/CNW_Satzung_aktuell.pdf).			
3.	Qualifikationsziele/Lernergebnisse/Kompetenzen Die Studierenden <ul style="list-style-type: none"> • beherrschen wesentliche mathematische Denkweisen als Grundlagen der Informatik; • können formal definieren, argumentieren und in Ansätzen modellieren; • können einfache Beweise (einschließlich Induktionsbeweise) eigenständig führen; • verstehen Logik als Grundlage korrekten Programmierens; • verstehen algebraische Denkweisen als formale Grundlage von Datenstrukturen <p>Durch die Übungen erarbeiten sie sich einen sicheren, präzisen und selbständigen Umgang mit den in den Vorlesungen behandelten Begriffen, Aussagen und Methoden; Die Studierenden sind im analytischen Denken geschult; sie sind in der Lage, abstrakte Strukturen zu erkennen und mathematische Probleme phantasievoll zu bearbeiten.</p>			
4.	Inhalte <ul style="list-style-type: none"> • Grundlagen formalen Denkens: Beweisen und Begründen • Beweistypen und -techniken • Grundlagen des Formalisierens: Logik und Mengenlehre • Logik: Aussagenlogik und Prädikatenlogik, Kalküle, informatische Anwendungen • Graphentheorie (grundlegende Begriffe); • Mengenlehre: Mengenoperationen, Relationen, Funktionen • Mächtigkeit von Mengen, elementare Kombinatorik, Anwendungen in der Wahrscheinlichkeitsrechnung • Grundlegende algebraische Konzepte • Ausgewählte Erweiterungen und Anwendungen 			
5.	Verwendbarkeit des Moduls B.Sc. Informatik, B.Ed. Informatik, M.Sc. Angewandte Bioinformatik			
6.	Empfohlene Voraussetzung(en) für die Teilnahme Brückenkurs Mathematik			
7.	Zugangsvoraussetzung(en)			
8.	Leistungsüberprüfungen 8.1. Aktive Teilnahme an b) 8.2. Studienleistung(en) 8.3. Modulprüfung Klausur (120 Minuten)			
9.	Stellenwert der Note in der Endnote bei Ein-Fach-Studiengängen bzw. Fachnote bei Mehr-Fächer-Studiengängen Note geht mit 8 LP in die Abschlussnote ein.			
10.	Häufigkeit des Angebots jedes Semester			

Modul 1: Formale Grundlagen der Informatik

11. Modulbeauftragte oder -beauftragter sowie hauptamtlich Lehrende

Prof. Dr. E. Althaus, Institut für Informatik
Studiengangsbeauftragte(r) des Instituts für Mathematik

12. Sonstige Informationen

Modul 2: Grundlagen der Fachdidaktik Informatik

Modul-Kennnummer (JOGU-StINe)	Arbeitsaufwand (workload)	Moduldauer (laut Studienverlaufsplan)	Regelsemester (laut Studienverlaufsplan)	Leistungspunkte (LP)
	300 h	2 Semester	5.+6. o. 4.+5. Semester	10 LP
1.	Lehrveranstaltungen/Lehrformen	Kontaktzeit	Selbststudium	Leistungspunkte
	a) Fachdidaktik I – Vorlesung (P)	3 SWS/31,5 h	89 h	4 LP
	b) Übung zur Vorlesung a) (P)	2 SWS/21 h	39 h	2 LP
	c) Fachdidaktik I - Hauptseminar	2 SWS/21 h	99 h	4 LP
2.	Gruppengrößen Gemäß aktueller Satzung über die Betreuungsrelationen von Lehrveranstaltungen in Bachelor- und Masterstudiengängen und zur Festsetzung der Normwerte für den Ausbildungsaufwand (Curricularnormwerte) der Johannes Gutenberg-Universität Mainz (http://www.uni-mainz.de/studlehr/ordnungen/CNW_Satzung_aktuell.pdf).			
3.	Qualifikationsziele/Lernergebnisse/Kompetenzen Die Studierenden <ul style="list-style-type: none"> kennen die Bildungsziele des Fachs Informatik und der damit verknüpften MINT-Fächer und können diese im Kontext grundständigen Informatikunterrichts sowie im Kontext weiterer Fächer (z. B. im Rahmen der Informations- und Kommunikationstechnischen Grundbildung) begründen und ihre Legitimation und Entwicklung im gesellschaftlichen und historischen Kontext darstellen und reflektieren kennen fachdidaktische Theorien und die fachdidaktische Forschung für Lehren und Lernen und können diese beispielhaft auf konkrete Unterrichtssituationen übertragen kennen schulische und außerschulische Anwendungsfelder der Informatik und können diese kritisch analysieren sowie daran den Bildungsauftrag des Fachs Informatik erläutern strukturieren Informatikunterricht methodisch sinnvoll können Informatikunterricht schülerzentriert gestalten erläutern den Bildungsauftrag des Fachs Informatik; kennen die Lerninhalte im Informatikunterricht verschiedener Schulstufen; bereiten diese unter Berücksichtigung fachdidaktischer und lernpsychologischer Prinzipien sowie inklusiver Konzepte altersgerecht und binnendifferenziert auf; kennen geeignete Software-Werkzeuge zur Unterstützung von Lehr-/Lern-Prozessen; strukturieren Unterrichtseinheiten methodisch sinnvoll. 			
4.	Inhalte <ul style="list-style-type: none"> Lerninhalte im Informatikunterricht Kennen und Anwenden der Konzepte des Lehrens und Lernens im Fach Informatik Beherrschen der Denkweisen und Methoden der Informatik und ihre Übertragung auf den Schulunterricht Konzeption und Gestaltung von Informatikunterricht, insbesondere Legitimierung von Informatikunterricht: Beitrag des Fachs zur Allgemeinbildung, Leitlinien informatischer Bildung, Ziele des Informatikunterrichts, Grundsätze und Standards des Informatikunterrichts Zentrale Ideen und Werkzeuge im Informatikunterricht Lehr-/Lern-Prozesse im Informatikunterricht Paradigmen der informatischen Modellierung, insbesondere imperative/objektorientierte, funktionale sowie wissensbasierte Programmierparadigmen an schulpraktischen Beispielen Werkzeuge zur Unterstützung der Lehr-/Lern-Prozesse im Informatikunterricht Genetischer Vermittlungsansatz für die Informatik Grenzen algorithmisch arbeitender Systeme im Unterricht Projektmethode Sichtbildung als informatisches Modellierungswerkzeug am Beispiel von Datenbanken 			
5.	Verwendbarkeit des Moduls B.Ed. Informatik			
6.	Empfohlene Voraussetzung(en) für die Teilnahme Module 3, 4, 6 und 8			
7.	Zugangsvoraussetzung(en)			
8.	Leistungsüberprüfungen			

Modul 2: Grundlagen der Fachdidaktik Informatik

	8.1. Aktive Teilnahme
	8.2. Studienleistung(en)
	8.3. Modulprüfung
	<ul style="list-style-type: none">- mündl. Prüfung (30 Min.) zu a) und b)- Portfolio zu c) bestehend aus Vortrag (30-45 Min.) + schriftlicher Ausarbeitung. Die Modulteilprüfungen werden entsprechend der zugeordneten LP gewichtet (6:4).
9.	Stellenwert der Note in der Endnote bei Ein-Fach-Studiengängen bzw. Fachnote bei Mehr-Fächer-Studiengängen Note geht mit 10 LP in die Endnote ein.
10.	Häufigkeit des Angebots jährlich
11.	Modulbeauftragte oder -beauftragter sowie hauptamtlich Lehrende Prof. Dr. Jens Gallenbacher
12.	Sonstige Informationen

Modul 3: Grundlagen der Programmierung

Modul-Kennnummer (JOGU-StINe)	Arbeitsaufwand (workload)	Moduldauer (laut Studienverlaufsplan)	Regelsemester (laut Studienverlaufsplan)	Leistungspunkte (LP)
	330 h	2 Semester	1. u. 2. Semester	11 LP
1.	Lehrveranstaltungen/Lehrformen	Kontaktzeit	Selbststudium	Leistungspunkte
	a) Einführung in die Programmierung – Vorlesung (P)	2 SWS/21 h	65 h	3 LP
	b) Übung zur Vorlesung a) (P)	2 SWS/21 h	40 h	2 LP
	c) Programmierpraktikum zu a) (P)	2 SWS/21 h	15 h	1 LP
	d) Einführung in die Softwareentwicklung – Vorlesung (P)	2 SWS/21 h	65 h	3 LP
	e) Übungen zur Vorlesung d) (P)	2 SWS/21 h	40 h	2 LP
2.	Gruppengrößen			
	Gemäß aktueller Satzung über die Betreuungsrelationen von Lehrveranstaltungen in Bachelor- und Masterstudiengängen und zur Festsetzung der Normwerte für den Ausbildungsaufwand (Curricularnormwerte) der Johannes Gutenberg-Universität Mainz (http://www.uni-mainz.de/studlehr/ordnungen/CNW_Satzung_aktuell.pdf).			
3.	Qualifikationsziele/Lernergebnisse/Kompetenzen			
	Die Studierenden			
	<ul style="list-style-type: none"> kennen die unterschiedlichen Programmierparadigmen haben vertiefte Kenntnisse in einer imperativen und einer objektorientierten Programmiersprache kennen grundlegende Modellierungskonzepte. 			
4.	Inhalte			
	<p>Einführung in die Programmierung:</p> <ul style="list-style-type: none"> Einführung leicht zu erlernende imperative Programmiersprache (z.B. Python) Variablenbegriff, Kontrollstrukturen, Unterprogramme/Funktionen Datentypen und zusammengesetzte Datenstrukturen (Arrays, Records/Klassen, Referenzen) Einfacher Algorithmen, z.B. zum Suchen und Sortieren Rekursion Pre- und Postconditions Asymptotische Komplexität (Grundkonzepte, einfache Beispiele) Softwaretests und Debugging Leitlinien für sauberes Programmieren (z.B. Schnittstellen, Modularisierung, Namen, Dokumentation) <p>Einführung in die Softwareentwicklung:</p> <ul style="list-style-type: none"> Einführung einer statisch typisierten, maschinennahen Programmiersprache (z.B. C++) Statische vs. dynamische Typisierung Maschinennahe Programmiertechniken und Sprachkonstrukte (Zeiger, Speicherverwaltung, etc.) Generische Algorithmen und Typen (in C++: Templates) Objektorientierte Programmierung <ul style="list-style-type: none"> Dynamisches OOP („Smalltalk“-Stil) Statische Typisierung für OOP („C++/JAVA“-Stil: Subtyping, virtuelle Methoden) Grundlagen objektorientierter Programmierung (Klassen, Schnittstellen, Geheimnisprinzip/ADTs, Vererbung, dynamische Bindung, ...) UML (Klassendiagramme, Objektdiagramme) Entwurfsmuster auf Klassenebene (z.B. Decorator, Observer, Visitor, Iterator, oder MVC) Architekturmuster aus der Praxis am Vorbild von Standardbibliotheken (z.B. Client-Server, verschiedene Ansätze für GUI-Bibliotheken, Ereignisorientierte Architekturen) Programmiertechniken (Nebenläufigkeit, GUIs, I/O). 			
5.	Verwendbarkeit des Moduls			
	B.Ed. Informatik, B.Sc. Informatik, M.Sc. Angewandte Bioinformatik, B.Sc. Mathematik, B.Sc. Physik, B.Sc. Wirtschaftspädagogik			
6.	Empfohlene Voraussetzung(en) für die Teilnahme			
7.	Zugangsvoraussetzung(en)			

Modul 3: Grundlagen der Programmierung

8.	Leistungsüberprüfungen <i>8.1. Aktive Teilnahme</i> <i>b), c) und e)</i> <i>8.2. Studienleistung(en)</i> zu a) und b): Klausur (180 Min.) <i>8.3. Modulprüfung</i> Klausur (120 Min.)
9.	Stellenwert der Note in der Endnote bei Ein-Fach-Studiengängen bzw. Fachnote bei Mehr-Fächer-Studiengängen Note geht mit 11 LP in die Endnote ein.
10.	Häufigkeit des Angebots jedes Semester
11.	Modulbeauftragte oder -beauftragter sowie hauptamtlich Lehrende Prof. Dr. Michael Wand / Prof. Dr. Andreas Hildebrandt
12.	Sonstige Informationen

Modul 4: Algorithmen und Datenstrukturen

Modul-Kennnummer (JOGU-StINe)	Arbeitsaufwand (workload)	Moduldauer (laut Studienverlaufsplan)	Regelsemester (laut Studienverlaufsplan)	Leistungspunkte (LP)
	270 h	1 Semester	3. o. 4. Semester	9 LP
1.	Lehrveranstaltungen/Lehrformen	Kontaktzeit	Selbststudium	Leistungspunkte
	a) Datenstrukturen u. effiziente Algorithmen - Vorlesung (P)	4 SWS/42 h	138h	6 LP
	b) Übung zur Vorlesung a) (P)	2 SWS/21 h	69 h	3 LP
2.	Gruppengrößen Gemäß aktueller Satzung über die Betreuungsrelationen von Lehrveranstaltungen in Bachelor- und Masterstudiengängen und zur Festsetzung der Normwerte für den Ausbildungsaufwand (Curricularnormwerte) der Johannes Gutenberg-Universität Mainz (http://www.uni-mainz.de/studlehr/ordnungen/CNW_Satzung_aktuell.pdf).			
3.	Qualifikationsziele/Lernergebnisse/Kompetenzen Die Studierenden <ul style="list-style-type: none"> • kennen grundlegende Datenstrukturen, Algorithmen und grundlegende Modellierungskonzepte; • entwickeln ein Verständnis für die Wechselwirkung zwischen Algorithmus und Datenstruktur; • können Softwaremodule modellieren, entwerfen, implementieren und die Qualität der Ergebnisse bewerten; • setzen mathematische Methoden zum Korrektheitsbeweis und zur Effizienzanalyse ein und können die Qualität von Algorithmen einschätzen. 			
4.	Inhalte <ul style="list-style-type: none"> • Grundlegende Datenstrukturen, abstrakte Datentypen und ihre Realisierung durch Datenstrukturen (Listen, Bäume) und fortgeschrittene Datenstrukturen (balancierte Bäume, Hash-Tabellen) • Grundlegende Algorithmen (z.B. Suchen und Sortieren, Graphenalgorithmen, Flussprobleme) • Algorithmische Prinzipien (Teile und herrsche, systematische Suche, Greedy Strategien) • Entwurf einfacher Algorithmen • Analysetechniken: Analyse randomisierter Algorithmen, amortisierte Analysen • Effizienzanalyse von Algorithmen • Zeit- und Platzkomplexität von Algorithmen • Asymptotisches Wachstum von Komplexität • spezielle Algorithmen (z. B. für Geometrie-, Codierungs-, Kommunikations- und Optimierungsprobleme, kryptografische Algorithmen) 			
5.	Verwendbarkeit des Moduls B.Ed. Informatik, B.Sc. Informatik, M.Sc. Angewandte Bioinformatik, B.Sc. Mathematik, B.Sc. Physik,			
6.	Empfohlene Voraussetzung(en) für die Teilnahme Modul 1 und Modul 3			
7.	Zugangsvoraussetzung(en)			
8.	Leistungsüberprüfungen 8.1. Aktive Teilnahme b), 8.2. Studienleistung(en) 8.3. Modulprüfung Klausur (120 Min.)			
9.	Stellenwert der Note in der Endnote bei Ein-Fach-Studiengängen bzw. Fachnote bei Mehr-Fächer-Studiengängen Note geht mit 9 LP in die Endnote ein.			
10.	Häufigkeit des Angebots jährlich			
11.	Modulbeauftragte oder -beauftragter sowie hauptamtlich Lehrende			

Modul 4: Algorithmen und Datenstrukturen

Prof. Dr. Ernst Althaus / Prof. Dr. Elmar Schömer

12. Sonstige Informationen

Modul 5: Programmierpraktikum

Modul-Kennnummer (JOGU-StINe)	Arbeitsaufwand (workload)	Moduldauer (laut Studienverlaufsplan)	Regelsemester (laut Studienverlaufsplan)	Leistungspunkte (LP)
	90 h	1 Semester	4. o. 5. Semester	3 LP
1.	Lehrveranstaltungen/Lehrformen	Kontaktzeit	Selbststudium	Leistungspunkte
	a) Programmierprojekt – Praktikum (P)	2 SWS/ 21h	69 h	3 LP
2.	Gruppengrößen Gemäß aktueller Satzung über die Betreuungsrelationen von Lehrveranstaltungen in Bachelor- und Masterstudiengängen und zur Festsetzung der Normwerte für den Ausbildungsaufwand (Curricularnormwerte) der Johannes Gutenberg-Universität Mainz (http://www.uni-mainz.de/studlehr/ordnungen/CNW_Satzung_aktuell.pdf).			
3.	Qualifikationsziele/Lernergebnisse/Kompetenzen Die Studierenden <ul style="list-style-type: none"> • können eine Anwendung entwerfen und implementieren; • können Softwaretests durchführen 			
4.	Inhalte Praktische Einübung der Inhalte des Moduls 4			
5.	Verwendbarkeit des Moduls B.Ed. Informatik			
6.	Empfohlene Voraussetzung(en) für die Teilnahme Modul 3 und Modul 4			
7.	Zugangsvoraussetzung(en)			
8.	Leistungsüberprüfungen 8.1. <i>Aktive Teilnahme</i> a) 8.2. <i>Studienleistung(en)</i> Portfolio bestehend aus verschiedenen Präsentationen während des Projektzeitraums. 8.3. <i>Modulprüfung</i>			
9.	Stellenwert der Note in der Endnote bei Ein-Fach-Studiengängen bzw. Fachnote bei Mehr-Fächer-Studiengängen Note geht nicht in die Endnote ein.			
10.	Häufigkeit des Angebots jährlich			
11.	Modulbeauftragte oder -beauftragter sowie hauptamtlich Lehrende Prof. Dr. Elmar Schömer / Prof. Dr. Ernst Althaus			
12.	Sonstige Informationen			

Modul 6: Informationssysteme

Modul-Kennnummer (JOGU-StINe)	Arbeitsaufwand (workload)	Moduldauer (laut Studienverlaufsplan)	Regelsemester (laut Studienverlaufsplan)	Leistungspunkte (LP)
	180 h	1 Semester	4. o. 3. Semester	6 LP
1.	Lehrveranstaltungen/Lehrformen	Kontaktzeit	Selbststudium	Leistungspunkte
	a) Datenbanken – Vorlesung (P)	2 SWS / 21 h	69 h	3 LP
	b) Übung zur Vorlesung a) (P)	2 SWS / 21 h	69 h	3 LP
2.	Gruppengrößen Gemäß aktueller Satzung über die Betreuungsrelationen von Lehrveranstaltungen in Bachelor- und Masterstudiengängen und zur Festsetzung der Normwerte für den Ausbildungsaufwand (Curricularnormwerte) der Johannes Gutenberg-Universität Mainz (http://www.uni-mainz.de/studlehr/ordnungen/CNW_Satzung_aktuell.pdf).			
3.	Qualifikationsziele/Lernergebnisse/Kompetenzen Die Studierenden <ul style="list-style-type: none"> • sind in der Lage relationale Datenbanken zu entwerfen, redundanzfrei zu machen, anzulegen und zu befragen. • können die theoretischen Grundlagen des relationalen Modells erklären: relationale Algebra und relationale Entwurfstheorie (Normalformen, funktionale und mehrwertige Abhängigkeiten, Dekomposition), • sind in der Lage die praktischen Aspekte in der Anwendung zu berücksichtigen, insbesondere die Nutzung von Indexstrukturen, die Optimierung von Anfragen und die Nutzung des Transaktionskonzepts • verstehen die Arbeitsweise relationaler Datenbankverwaltungssysteme; • konzipieren und realisieren den Einsatz eines solchen Systems; • setzen die standardisierte Datenbanksprache SQL ein. 			
4.	Inhalte <ul style="list-style-type: none"> • Datenmodellierung und Datenbankentwurf • Entity-Relationship-Modellierung • Anfragesprachen: Relationenalgebra, Standardsprache SQL • Strukturelle und domänenspezifische Integrität • Relationale Entwurfstheorie: Funktionale Abhängigkeiten, Normalformen • Transaktionsmanagement, Transaktionskonzept (ACID) • Formale Semantik von Anfragesprachen • Einführung und Grundbegriffe, Grundlagen der Informationssuche • Data-Mining • DBS-Architektur und DB-Pufferverwaltung • Effizienter Datenzugriff durch Indexe (B/B+-Bäume, Hashing, Bulkloading) • Aspekte von Big-Data Management (NoSQL, CAP Theorem, Eventual Consistency) 			
5.	Verwendbarkeit des Moduls B.Sc. Informatik, B.Ed. Informatik,			
6.	Empfohlene Voraussetzung(en) für die Teilnahme Modul 1 und Modul 3			
7.	Zugangsvoraussetzung(en)			
8.	Leistungsüberprüfungen 8.1. Aktive Teilnahme b) 8.2. Studienleistung(en) 8.3. Modulprüfung Klausur (120 Minuten)			
9.	Stellenwert der Note in der Endnote bei Ein-Fach-Studiengängen bzw. Fachnote bei Mehr-Fächer-Studiengängen Note geht mit 6 LP in die Endnote ein.			

Modul 6: Informationssysteme

10.	Häufigkeit des Angebots jährlich
11.	Modulbeauftragte oder -beauftragter sowie hauptamtlich Lehrende Prof. Dr. Stefan Kramer / Prof. Dr. P. Bouros
12.	Sonstige Informationen

Modul 7: Informatik und Gesellschaft

Modul-Kennnummer (JOGU-StINe)	Arbeitsaufwand (workload)	Moduldauer (laut Studienverlaufsplan)	Regelsemester (laut Studienverlaufsplan)	Leistungspunkte (LP)
	90 h	1 Semester	6. o. 3. Semester	3 LP
1.	Lehrveranstaltungen/Lehrformen	Kontaktzeit	Selbststudium	Leistungspunkte
	a) Informatik und Gesellschaft – Hauptseminar (P)	2 SWS / 21 h	70 h	3 LP
2.	Gruppengrößen Gemäß aktueller Satzung über die Betreuungsrelationen von Lehrveranstaltungen in Bachelor- und Masterstudiengängen und zur Festsetzung der Normwerte für den Ausbildungsaufwand (Curricularnormwerte) der Johannes Gutenberg-Universität Mainz (http://www.uni-mainz.de/studlehr/ordnungen/CNW_Satzung_aktuell.pdf).			
3.	Qualifikationsziele/Lernergebnisse/Kompetenzen Die Studierenden <ul style="list-style-type: none"> • wissen um die Wechselwirkungen zwischen der Informatik und der Gesellschaft; • kennen und beachten wesentliche Verhaltensregeln für Informatikerinnen und Informatiker; • verfügen über grundlegende Rechtskenntnisse und ein Rechtsbewusstsein im Umgang mit Informatiksystemen. 			
4.	Inhalte <ul style="list-style-type: none"> • Verantwortliches Handeln im Umgang mit Informatiksystemen • Rolle von Informatiksystemen für die gesellschaftliche und soziale Teilhabe • Einsatz von Symbolsystemen, die die Wahrnehmung und Kommunikation unterstützen und fördern, z.B. Morse-Code, Braille-Schrift, angemessene Gestaltung der Benutzungsoberfläche • Informationelle Selbstbestimmung • rechtliche Aspekte (z. B. Urheberrecht, Persönlichkeitsrecht, Plagiate) • Rolle von Informationssystemen für die gesellschaftliche und soziale Teilhabe • Richtlinien, Verhaltensregeln, Ethik • Datenschutz und IT-Sicherheit • Virtuelle Welten • Geschichtliche Entwicklungen der Informatik 			
5.	Verwendbarkeit des Moduls B.Ed. Informatik			
6.	Empfohlene Voraussetzung(en) für die Teilnahme			
7.	Zugangsvoraussetzung(en)			
8.	Leistungsüberprüfungen 8.1. <i>Aktive Teilnahme</i> 8.2. <i>Studienleistung(en)</i> 8.3. <i>Modulprüfung</i> Portfolio bestehend Vortrag (ca. 45 Min.) und schriftlicher Ausarbeitung (ca. 15 Seiten) zu einem gegebenen Thema.			
9.	Stellenwert der Note in der Endnote bei Ein-Fach-Studiengängen bzw. Fachnote bei Mehr-Fächer-Studiengängen Note geht mit 3 LP in die Endnote ein.			
10.	Häufigkeit des Angebots jährlich			
11.	Modulbeauftragte oder -beauftragter sowie hauptamtlich Lehrende Dr. Hans-Jürgen Schröder			
12.	Sonstige Informationen			

Modul 7: Informatik und Gesellschaft	

Modul 8: Grundlagen der technischen Informatik

Modul-Kennnummer (JOGU-StINe)	Arbeitsaufwand (workload)	Moduldauer (laut Studienverlaufsplan)	Regelsemester (laut Studienverlaufsplan)	Leistungspunkte (LP)
	150 h	1 Semester	2. Semester	5 LP
1.	Lehrveranstaltungen/Lehrformen	Kontaktzeit	Selbststudium	Leistungspunkte
	a) Technische Informatik - Vorlesung(P)	2 SWS/21 h	69 h	3 LP
	b) Übung zu a) (P)	2 SWS/21 h	39 h	2 LP
2.	Gruppengrößen Gemäß aktueller Satzung über die Betreuungsrelationen von Lehrveranstaltungen in Bachelor- und Masterstudiengängen und zur Festsetzung der Normwerte für den Ausbildungsaufwand (Curricularnormwerte) der Johannes Gutenberg-Universität Mainz (http://www.uni-mainz.de/studlehr/ordnungen/CNW_Satzung_aktuell.pdf).			
3.	Qualifikationsziele/Lernergebnisse/Kompetenzen Die Studierenden <ul style="list-style-type: none"> • verfügen über ein Grundverständnis für die Funktionsweise eines Einprozessor-Rechners; • kennen dessen grundlegende Struktur, wissen, wie ein Befehl interpretiert wird, und kennen einige Optimierungstechniken; • kennen die elektrotechnische Realisierung von Schaltungen sowie der Ein- und Ausgabe über Sensoren und Aktuatoren bei technischen Systemen; • kennen grundlegende Rechnerstrukturen (wie z.B. Rechnerarithmetik, Addierer, Multiplizierer, Multiplexer, PLAs) und haben damit die Fähigkeit zur Leistungsanalyse von Rechnern erworben; • sind in der Lage, die Elemente des Rechners zu entwerfen, kleinere Assemblerprogramme zu schreiben und wesentliche Funktionen eines Betriebssystems zu verstehen. 			
4.	Inhalte <ul style="list-style-type: none"> • Darstellung von Informationen durch Daten • Zahlendarstellungen und Rechnerarithmetik • Aufbau und Funktionsweise von Rechnern, Mikroarchitektur eines Prozessors • Befehlsinterpretation, Befehlsfließband • Speicherhierarchie; Ein-/Ausgabe • digitaltechnische und elektrotechnische Grundlagen (u. a. boolesche Algebra, Schaltalgebra, kombinatorische und sequenzielle Logik, Grundlagen von Schaltkreisen, Schaltnetze und deren Realisierung, Schaltwerke) • Assemblerprogrammierung und deren Anwendung zur Realisierung höherer Programmiersprachen • Binder und Lader, Unterbrechungsstrukturen und Synchronisation, Prozessverwaltung; • Ein-/Ausgabe (inkl. Sensor-/Aktuator-Systeme) • Hauptspeicherverwaltung, Dateiverwaltung, Schutzmechanismen • Grundlagen von Betriebssystemen 			
5.	Verwendbarkeit des Moduls B.Sc. Informatik, B.Ed. Informatik			
6.	Empfohlene Voraussetzung(en) für die Teilnahme			
7.	Zugangsvoraussetzung(en)			
8.	Leistungsüberprüfungen 8.1. Aktive Teilnahme b) 8.2. Studienleistung(en) 8.3. Modulprüfung Klausur (120 Minuten)			
9.	Stellenwert der Note in der Endnote bei Ein-Fach-Studiengängen bzw. Fachnote bei Mehr-Fächer-Studiengängen Note geht mit 5 LP in die Endnote ein.			

Modul 8: Grundlagen der technischen Informatik

10.	Häufigkeit des Angebots jedes Semester
11.	Modulbeauftragte oder -beauftragter sowie hauptamtlich Lehrende Prof. Dr. Bertil Schmidt, Prof. Dr. André Brinkmann
12.	Sonstige Informationen

Modul 9: Grundlagen der theoretischen Informatik

Modul-Kennnummer (JOGU-StINe)	Arbeitsaufwand (workload)	Moduldauer (laut Studienverlaufsplan)	Regelsemester (laut Studienverlaufsplan)	Leistungspunkte (LP)
	300 h	2 Semester	4./5. o. 5./6. Semester	10 LP
1.	Lehrveranstaltungen/Lehrformen	Kontaktzeit	Selbststudium	Leistungspunkte
	a) Formale Sprachen u. Berechenbarkeit-Vorlesung (P)	2 SWS/21 h	69 h	3 LP
	b) Übung zur Vorlesung a) (P)	2 SWS/21 h	39 h	2 LP
	c) Komplexitätstheorie – Vorlesung (P)	2 SWS/21 h	69 h	3 LP
	d) Übung zur Vorlesung c) (P)	2 SWS/21 h	39 h	2 LP
2.	Gruppengrößen Gemäß aktueller Satzung über die Betreuungsrelationen von Lehrveranstaltungen in Bachelor- und Masterstudiengängen und zur Festsetzung der Normwerte für den Ausbildungsaufwand (Curricularnormwerte) der Johannes Gutenberg-Universität Mainz (http://www.uni-mainz.de/studlehr/ordnungen/CNW_Satzung_aktuell.pdf).			
3.	Qualifikationsziele/Lernergebnisse/Kompetenzen Die Studierenden <ul style="list-style-type: none"> • verfügen über ein Verständnis für die Grundlagenfragen der Informatik; • kennen Automaten und formale Sprachen sowie deren Zusammenhänge; • kennen Verfahren zur Beurteilung der Berechenbarkeit und Entscheidbarkeit; • kennen Komplexitätsmaße und Methoden zur Bewältigung von Komplexität; • können mathematische Methoden zur Klärung von Grundlagenfragen der Informatik anwenden. 			
4.	Inhalte Formale Sprachen und Berechenbarkeit: <ul style="list-style-type: none"> • Formale Sprachen und Grammatiken, • endliche Automaten und Kellerautomaten, • Logikkalküle • Chomsky-Hierarchie • Turing-Maschinen, • Unentscheidbarkeit • Reduktion Komplexitätstheorie: <ul style="list-style-type: none"> • Algorithmusbegriff • Aufwandsabschätzung, • Klassen P und NP, • NP-Vollständigkeit • Algorithmen für NP-vollständige Probleme • Randomisierung • Einführung in die Kryptographie 			
5.	Verwendbarkeit des Moduls B.Sc. Informatik, B.Ed. Informatik, B.Sc. Mathematik			
6.	Empfohlene Voraussetzung(en) für die Teilnahme Modul 1			
7.	Zugangsvoraussetzung(en)			
8.	Leistungsüberprüfungen 8.1. Aktive Teilnahme b) und d) 8.2. Studienleistung(en) 8.3. Modulprüfung			

Modul 9: Grundlagen der theoretischen Informatik

	Klausur (120 Minuten)
9.	Stellenwert der Note in der Endnote bei Ein-Fach-Studiengängen bzw. Fachnote bei Mehr-Fächer-Studiengängen Note geht mit 10 LP in die Endnote ein.
10.	Häufigkeit des Angebots Jedes Semester
11.	Modulbeauftragte oder -beauftragter sowie hauptamtlich Lehrende Prof. Dr. E. Althaus, Prof. Dr. Bertil Schmidt
12.	Sonstige Informationen