

Modulbezeichnung	Einführung in die Informatik		
Semester	1		
Dauer	1 Semester		
Art	Pflichtfach		
ECTS-Punkte	2,5		
Studentische Arbeitsbelastung	70 h Kontaktzeit + h Selbststudium		
Voraussetzungen (laut BPO)			
Empf. Voraussetzungen			
Prüfungsform und -dauer	Klausur 1,5 h Schriftliche Dokumentation		
Lehr- und Lernmethoden	Vorlesung		
Modulverantwortlicher	G. Brands		
Qualifikationsziele	Die Studenten kennen die wesentlichen Komponenten eines Rechnersystems und ihre Aufgaben. Sie sind mit den grundlegenden Funktionsweisen der Komponenten vertraut. Sie kennen die wesentlichen Softwarekomponenten und deren Grundfunktionen. Sie kennen die Zahlenmodelle und die damit verbundenen Fehlerquellen und können die Qualität von Rechenergebnissen abschätzen. Sie kennen die Basisprotokolle der Netzwerkverbindungen zwischen Rechnern und können deren Einsatzkonfiguration nebst Risikoabschätzungen planen.		
Lehrinhalte	Die Studenten werden schrittweise an die notwendige Denkweise bei der Programmierung herangeführt, die in anderen Modulen vertieft wird. Die Komponenten und ihre Arbeitsweise und Arbeitsteilung untereinander wird vorgestellt, beispielsweise Festplatten, CPU, Hauptspeicher, Bildschirmspeicher usw. Zahlenmodelle und das Entstehen von Rundungsfehlern und deren Fortpflanzung wird in Übungen untersucht. Die notwendigen Basisprotokolle für den Betrieb von Rechnern in einfachen Netzwerktopologien sowie die korrekte Konfiguration werden diskutiert.		
Literatur			
Lehrveranstaltungen			
Dozent	Titel der Lehrveranstaltung	SWS	
G. Brands	Einführung in die Informatik	2	

Modulbezeichnung	Hardwaregrundlagen	
Semester	1	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen		
Verwendbarkeit	BaMT	
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	R. Wenzel	
Qualifikationsziele	Die Studierenden lernen elementare Grundlagen der analogen und digitalen Elektronik kennen. Sie sind in der Lage, sowohl passive als auch aktive Bauelemente anzuwenden und die zugehörige Meßtechnik einzusetzen. Dabei wird auch der Unterschied zwischen Theorie und Praxis an ausgewählten Beispielen erläutert und nachgewiesen. Schaltungsanalyse- und synthese dienen zum komplexen Verständnis elektronischer Baugruppen.	
Lehrinhalte	Wichtige Bauelemente, wie z.B. Widerstände, Dioden und Transistoren werden hinsichtlich ihres Aufbaus, ihrer Funktionsweise und ihrer Anwendung beschrieben. Einfache Netzwerke werden dabei dimensioniert, aufgebaut und bezüglich ihres elektrischen Verhaltens untersucht. Digitale Grundfunktionen und kombinatorische Schaltungen werden anhand von Beispielen beschrieben und ebenfalls getestet.	
Literatur	Beuth, K.:Bauelemente (Elektronik 2), Vogel, 2010 Beuth, K.:Digitaltechnik (Elektronik 4),Vogel, 2010	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
R. Wenzel	Hardwaregrundlagen	3
D. Rabe	Praktikum Hardwaregrundlagen	1

Modulbezeichnung	Mathematik 1	
Semester	1	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	7,5	
Studentische Arbeitsbelastung	105 h Kontaktzeit + 120 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen		
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Übung	
Modulverantwortlicher	G. von Cölln	
Qualifikationsziele	Die Studierenden kennen grundlegende Begriffe und Methoden aus der linearen Algebra, Analysis und diskreten Mathematik.	
Lehrinhalte	Themen der linearen Algebra, Analysis und diskreten Mathematik werden behandelt und das Wissen in Übungen wiederholt und vertieft. Stichworte zu den Inhalten sind: Funktionen, Grenzwerte, Differentialrechnung, Mengen und Relationen, Aussagenlogik, Analytische Geometrie, Matrizen	
Literatur	Teschl, Teschl: Mathematik für Informatiker Band 1 und 2, Springer, 2006. Socher, Mathematik für Informatiker, Hanser, 2011	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
G. von Cölln, M. Schiemann-Lillie	Mathematik 1	4
G. von Cölln, M. Schiemann-Lillie, D. Rabe, J. Wiebe	Übung Mathematik 1	2

Modulbezeichnung	Algorithmen und Datenstrukturen	
Semester	2	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Java 1	
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	G. Totzauer	
Qualifikationsziele	Die Studierenden kennen häufig verwendete Algorithmen mit ihren dazu gehörigen Datenstrukturen und können sie an Beispielen per Hand veranschaulichen. Sie kennen die Laufzeit und den Speicherbedarf der verschiedenen Algorithmen und können einfache Aufwandsanalysen selbständig durchführen. Sie sind in der Lage zu einer gegebenen Aufgabenstellung verschiedene Algorithmen effizient zu kombinieren und anschließend zu implementieren.	
Lehrinhalte	Häufig verwendete Algorithmen mit ihren dazu gehörigen Datenstrukturen werden vorgestellt und verschiedene Implementierungen bewertet. Stichworte sind: Listen, Bäume, Mengen, Sortierverfahren, Graphen und Algorithmenentwurfstechniken. Es wird besonderer Wert auf die Wiederverwendbarkeit der Implementierungen für unterschiedliche Grunddatentypen gelegt.	
Literatur	Heun, V.: Grundlegende Algorithmen, Vieweg, 2000. Sedgewick, R.: Algorithmen in Java, 3. überarbeitete Auflage, Pearson Studium, 2003.	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
G. Totzauer	Algorithmen und Datenstrukturen	2
G. Totzauer	Praktikum Algorithmen und Datenstrukturen	2

Modulbezeichnung	C/C++	
Semester	2	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Java 1, Einführung in die Informatik, Mathematik 1	
Prüfungsform und -dauer	Klausur 1,5 h Erstellung und Dokumentation von Rechnerprogrammen	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	G. Brands	
Qualifikationsziele	Die Studenten kennen die unterschiedlichen Datenspeichermodelle und wissen sie sicher einzusetzen. Sie kennen die hohe Typsicherheit und die feineren Steuerungsmöglichkeiten von C++ gegenüber anderen Sprachen sowie die Mechanismen der Operatorüberladung. Sie kennen das grundlegende Musterklassenkonzept und wissen um die sich daraus ergebende Möglichkeit der Entwicklung von Compileralgorithmen.	
Lehrinhalte	Anhand des Aufbaus einfacher Programme werden die grundlegenden Unterschiede und Erweiterungen zum Java-Konzept vorgestellt und die spezifischen Vokabeln der C++ Sprache und ihre Bedeutung erklärt. Speicherkonzepte, insbesondere Zeigervariablen, und der korrekte Umgang mit ihnen werden diskutiert. Die sich aus der Operatorladung und der Definition von Musterklassen (Templates) ergebenden Programmiermöglichkeiten werden an mathematischen Modellen und an den Standardbibliotheken demonstriert.	
Literatur	Gilbert Brands, Das C++ Kompendium, Springer 2010	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
G. Brands	C/C++	2
G. Brands	Praktikum C/C++	2

Modulbezeichnung	Java 2	
Semester	2	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Java 1	
Verwendbarkeit	BaMT	
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	F. Rump	
Qualifikationsziele	Die Studierenden sollen eine konkrete Problemstellung analysieren und algorithmisch lösen können. Sie kennen wichtige Java-Bibliotheken und können diese für konkrete Aufgabenstellungen anwenden. Die Programme werden auf Basis aktueller Werkzeuge erstellt und getestet. Die Studierenden verstehen das Verfahren der testgetriebenen Entwicklung und können dieses für kleine Beispiele anwenden.	
Lehrinhalte	Auf Basis der in "Java 1" gelegten Grundlagen werden weitergehende Konzepte der objektorientierten Programmierung vorgestellt und die Verwendung objektorientierter Bibliotheken vertieft. Behandelt werden u.a. Datenströme und Dateizugriff, Threads, Netzwerkprogrammierung, Unit-Tests, graphische Benutzungsoberflächen mit vorgegebenen Komponenten und Ereignisverarbeitung. Typische Programmstrukturen werden anhand gängiger Entwurfs- und Architekturmuster (z.B. Model-View-Controller) erläutert.	
Literatur	Ratz, D.; Scheffler, J.; Seese, D.; Wiesenberger, J.: Grundkurs Programmieren in Java, Hanser, 2010. Krüger, G.; Stark, T.: Handbuch der Java-Programmierung, Addison-Wesley, 2009.	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
F. Rump	Java 2	3
F. Rump	Praktikum Java 2	1

Modulbezeichnung	Mathematik 2	
Semester	2	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	7,5	
Studentische Arbeitsbelastung	105 h Kontaktzeit + 120 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen		
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Übung	
Modulverantwortlicher	G. von Cölln	
Qualifikationsziele	Die Studierenden kennen grundlegende Begriffe und Methoden aus der linearen Algebra, Analysis und diskreten Mathematik.	
Lehrinhalte	Themen der linearen Algebra, Analysis und diskreten Mathematik werden behandelt und das Wissen in Übungen wiederholt und vertieft. Stichworte zu den Inhalten sind: Folgen und Reihen, Integralrechnung, Funktionen in Parameterdarstellung, numerische Verfahren.	
Literatur	Teschl, Teschl: Mathematik für Informatiker Band 1 und 2, Springer, 2006. Socher, Mathematik für Informatiker, Hanser, 2011	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
G. von Cölln, M. Schiemann-Lillie	Mathematik 2	4
G. von Cölln, M. Schiemann-Lillie, D. Rabe, J. Wiebe	Übung Mathematik 2	2

Modulbezeichnung	Theoretische Informatik	
Semester	2	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Einführung in die Informatik, Mathematik 1	
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	C. Koch	
Qualifikationsziele	<p>Ziel des Kurses ist das Vermitteln von Grundkonzepten der Theoretischen Informatik. Die Studierenden sollen die grundlegenden Begriffe, Konzepte und Methoden endlicher Automaten und Grammatiken kennenlernen und selbständig Automaten für bestimmte Problemstellungen entwickeln können.</p> <p>Weiterhin beherrschen die Studierenden die verschiedenen Transformationen, können den Beweis der Nicht-Regularität einer Sprache führen und haben den Zusammenhang zwischen Automaten und Grammatiken erarbeitet.</p>	
Lehrinhalte	<p>Stichworte sind: Endliche Automaten (DEA, NEA und NEA mit epsilon-Übergängen), Kellerautomaten, reguläre Ausdrücke, Transformationen und Minimierung (NEA nach DEA, NEA/eps nach NEA, regulärer Ausdruck nach NEA/eps), reguläre und nicht-reguläre Sprachen, Grammatiken und kontextfreie Sprachen</p>	
Literatur	<p>Socher, R.: Theoretische Grundlagen der Informatik, Carl Hanser Verlag München, 2008</p> <p>Hedtstück, U.: Einführung in die Theoretische Informatik, Oldenburger Wissenschaftsverlag, 2007</p>	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
C. Koch	Theoretische Informatik	3
C. Koch	Praktikum Theoretische Informatik	1

Modulbezeichnung	Hardwarenahe Programmierung	
Semester	3	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)	Einführung in die Informatik, Hardware-Grundlagen	
Empf. Voraussetzungen	C/C++	
Verwendbarkeit	BaE	
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	C. Koch	
Qualifikationsziele	Die Studierenden sollen das Zusammenwirken von Software mit der Hardware eines Rechners verstehen und hieraus die Struktur einer Assemblersprache als auch ihrer wesentlichen Fähigkeiten ableiten können. Sie kennen hardwarespezifische Grundkonzepte und nutzen diese als Voraussetzung für effizientes Programmieren in höheren Programmiersprachen.	
Lehrinhalte	Das Modul zielt auf die Vermittlung folgender Lehrinhalte: Die generelle Architektur eines Mikroprozessors und sein Zusammenwirken mit dem Speicher und der Rechnerperipherie. Die Architektur einer Assemblersprache im Vergleich mit höheren Programmiersprachen als auch die eingehende Besprechung des Befehlssatzes der ausgewählten Assemblersprache (i8086-Architektur). Weitere Stichworte sind: Indirekte Adressierung, Unterprogrammtechnik und Interruptsystem als Basis des Programmierens in allen höheren Programmiersprachen.	
Literatur	Backer, R.: Programmiersprache Assembler, Rowohlt Hamburg, 2007 Patterson, D.A.:Rechnerorganisation und -entwurf, Elsevier München, 2005	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
C. Koch	Hardwarenahe Programmierung	3
C. Koch	Praktikum Hardwarenahe Programmierung	1

Modulbezeichnung	Internet-Technologien	
Semester	4	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen		
Prüfungsform und -dauer	Klausur 1,5h oder mündliche Prüfung	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	F. Rump	
Qualifikationsziele	Die Studierenden kennen verschiedene Protokolle, Architekturen und Techniken für moderne Internet-Anwendungen. Sie sind in der Lage, unterschiedliche Möglichkeiten zur Implementierung von Internet-Anwendungen einzuschätzen und selbst mit einer Auswahl an Techniken Internet-Anwendungen mit Datenbankbindung zu entwickeln.	
Lehrinhalte	Die Veranstaltung gibt eine Einführung in wichtige Protokolle, Architekturen und Techniken für moderne Internet-Anwendungen auf Basis der Programmiersprache Java (u.a. HTTP, HTML, XML, Mehrschichtenarchitekturen, Servlets, JSP, JavaBeans). Anhand eines konkreten MVC-Frameworks (z.B. JavaServer Faces) wird die Implementierung professioneller Internet-Anwendungen mit Datenbankbindung vermittelt.	
Literatur	Wöhr, H.: Web-Technologien: Konzepte - Programmiermodelle - Architekturen, dpunkt.verlag, 2004. Müller, B.: Java Server Faces 2.0 - Ein Arbeitsbuch für die Praxis, Hanser, 2010.	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
F. Rump	Internet-Technologien	3
F. Rump	Praktikum Internet-Technologien	1

Modulbezeichnung	Kryptologie	
Semester	4-6	
Dauer	2 Semester	
Art	Wahlpflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Mathematik 1, Mathematik 2, Mathematik 3, C/C++	
Prüfungsform und -dauer	Klausur 1,5 h Mündliche Präsentation und schriftliche Dokumentation	
Lehr- und Lernmethoden	Vorlesung, Studentische Arbeit	
Modulverantwortlicher	G. Brands	
Qualifikationsziele	Die Studenten kennen den Aufbau der wesentlichen Basisalgorithmen für Einweg-, symmetrische und asymmetrische Verschlüsselung sowie die wesentlichen bekannten Angriffsmethoden. Sie können anwendungsbedingt Hybridverfahren bewerten bzw. geeignete Hybridverfahren für bestimmte Anwendungszwecke auswählen. Sie kennen geeignete Algorithmen zur Implementation von Verfahren auf Rechnern. Sie kennen unterschiedliche Authentifizierungsmethoden und haben wesentliche in der Praxis eingesetzte Mechanismen experimentell erprobt.	
Lehrinhalte	Hashalgorithmen der aktuellen und der zukünftigen Generation, ihre Einsatzbereiche und Gefährdungen werden vorgestellt, desgleichen symmetrische Verschlüsselungsverfahren. Die mathematischen Grundlagen und ihre Übersetzung in Computeralgorithmen für symmetrische Verfahren werden ausführlich bis hin zu komplexeren Authentifizierungen diskutiert. Praktische Einsatzbereiche wie Kerberos, Plattenverschlüsselung und PKI werden anhand von Einsatzszenarien vorgestellt und in Übungen und Eigenarbeit der Studenten vertieft.	
Literatur	Gilbert Brands, Verschlüsselungsalgorithmen, Vieweg 2001 Gilbert Brands, Verschlüsselung, Theorie und Praxis, voraussichtlich 2011	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
G. Brands	Kryptologie, Vorlesung und Seminar	4

Modulbezeichnung	Rechnerarchitekturen	
Semester	4	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Hardware Grundlagen, Einführung Informatik, Hardwarenahe Programmierung	
Verwendbarkeit	BaE	
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung	
Modulverantwortlicher	G. von Cölln	
Qualifikationsziele	Die Studierenden verfügen über ein fundiertes, anwendungsorientiertes Wissen über den prinzipiellen Aufbau und die Arbeitsweise von Computern. Sie kennen die wesentlichen Komponenten und deren Zusammenwirken. Die Studierenden können die Leistungsfähigkeit von Computern beurteilen und sind in der Lage diese zu optimieren. Die Studierenden können die grundlegenden Konzepte moderner Computer in anderen technischen Systemen wieder erkennen bzw. diese zur Lösung eigener Aufgabenstellungen anwenden.	
Lehrinhalte	Aufbau und Funktionen von Computern werden vorgestellt. Zu Grunde liegenden Konzepte werden dargestellt und hinsichtlich verschiedener Kriterien bewertet. Stichworte sind: Grundlegende Begriffe, Funktion und Aufbau von Computern, Maßnahmen zur Leistungssteigerung, Speicherhierarchien, virtuelle Speicherverwaltung. Es wird besonderer Wert auf die grundlegenden Konzepte sowie auf die Übertragbarkeit auf andere Problemstellungen hingewiesen.	
Literatur	Patterson, Hennessy: Rechnerorganisation und -entwurf, Spektrum Akademischer Verlag, 3. Aufl. 2005 Tanenbaum, Andrew, S.: Computerarchitektur, Pearson Studium, 5. Aufl., 2005.	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
G. von Cölln	Rechnerarchitekturen	4

Modulbezeichnung	Parallele Systeme	
Semester	5	
Dauer	1 Semester	
Art	Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Rechnerarchitekturen	
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	G. von Cölln	
Qualifikationsziele	Die Studierenden verfügen über ein fundiertes, anwendungsorientiertes Wissen über den prinzipiellen Aufbau und die Arbeitsweise von parallelen Computersystemen. Sie kennen die wesentlichen Konzepte der Parallelverarbeitung auf verschiedenen Ebenen und deren Realisierung. Die Studierenden kennen die Einsatzgebiete und Grenzen der Leistungssteigerung durch Parallelverarbeitung. Sie können an einfachen Beispielen die Problematik der parallelen Programmierung erschließen und mit aktuellen Entwicklungswerkzeugen praktisch umsetzen.	
Lehrinhalte	Konzepte der Parallelverarbeitung auf verschiedenen Ebenen werden vorgestellt und bewertet. Entwicklungsmethoden und Werkzeuge zur parallelen Programmierung werden vorgestellt und an praktischen Beispielen angewendet. Stichworte sind: Konzepte und Organisationen zur Parallelverarbeitung, Superskalare Rechner, SMP und MPP, Speicherkonzepte, Entwicklungswerkzeuge und Sprachen.	
Literatur	Patterson, Hennessy: Rechnerorganisation und -entwurf, Spektrum Akademischer Verlag, 2005 Rauber, Rüniger: Parallele Programmierung, Springer, 2010	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
G. von Cölln	Parallele Systeme	3
G. von Cölln	Praktikum Parallele Systeme	1

Modulbezeichnung	HW/SW-Codesign	
Semester	6	
Dauer	1 Semester	
Art	Pflichtfach Vertiefung Technische Informatik	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)	Rechnerarchitekturen, Hardwarenahe Programmierung	
Empf. Voraussetzungen	C/C++, Digitaltechnik, Mikrocomputertechnik	
Verwendbarkeit	BaE	
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	C. Koch	
Qualifikationsziele	Ziel der Veranstaltung ist die Zusammenführung der zunächst im Studium getrennten Betrachtung von Hardware- und Software-Systemen zum Aufbau und Entwurf moderner eingebetteter Systeme. Die Studierenden erwerben hierbei weiterführenden Kenntnisse und Methoden hinsichtlich der Software- und Hardware-Entwicklung eingebetteter Systeme als auch deren Partitionierung.	
Lehrinhalte	Die Vorlesung Mikrocomputersysteme behandelt typische Zielarchitekturen und HW/SW-Komponenten von eingebetteten Standard-Systemen und System-on-Programmable-Chips (SoPC) sowie deren Entwurfswerkzeuge für ein Hardware/Software Codesign. Hierbei behandelte Zielarchitekturen und Rechenbausteine umfassen Mikrocontroller, DSP (VLIW, MAC), FPGAs, ASIC, System-on-Chip als auch Hybride Architekturen.	
Literatur	<p>Mahr, T: Hardware-Software-Codesign, Vieweg Verlag Wiesbaden, 2007.</p> <p>Patterson, D.A.: Rechnerorganisation und -entwurf, Elsevier München, 2005</p>	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
C. Koch	HW/SW-Codesign	3
C. Koch	Praktikum HW/SW-Codesign	1

Modulbezeichnung	Spezielle Verfahren der IT-Sicherheit	
Semester	6	
Dauer	1 Semester	
Art	Wahlpflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Kryptologie	
Prüfungsform und -dauer	Klausur 1,5 h Mündliche Prüfung Erstellung und Dokumentation von Rechnerprogrammen	
Lehr- und Lernmethoden	Vorlesung, Seminar	
Modulverantwortlicher	G. Brands	
Qualifikationsziele	Die Studenten kennen aktuelle Themen der IT-Sicherheit, können sich selbständig in auftretende Themen und Probleme einarbeiten und Lösungen nach Stand der Technik entwickeln und präsentieren.	
Lehrinhalte	Aktuelle Themen der IT-Sicherheit oder Forschungsthemen werden vorgestellt und Aufgabenstellungen formuliert, die von den Studenten selbständig zu bearbeiten und vorzutragen sind. Beispiele für Themenbereiche (stichwortartig): Quantencomputer, Quantenkryptografie, elektronischer Personalausweis, biometrische Identifizierung und Authentifizierung, komplexe Authentifizierungsprotolle, OpenDemocracy aus Sicht der Sicherheitstechnik, aktuelle Angriffsmethoden auf Verfahren	
Literatur	Gilbert Brands, IT-Sicherheitsmanagement, Springer 2006 Gilbert Brands, Einführung in die Quanteninformatik, ? 2010/2011	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
G. Brands	Seminar Spezielle Verfahren der IT-Sicherheit	4

Modulbezeichnung	Autonome Systeme	
Dauer	1 Semester	
Art	Wahlpflichtfach	
ECTS-Punkte	2	
Studentische Arbeitsbelastung	30 h Kontaktzeit + 40 h Selbststudium	
Voraussetzungen (laut BPO)	Einführung in die Informatik	
Empf. Voraussetzungen	Programmieren 1, Programmieren 2, Algorithmen und Datenstrukturen	
Verwendbarkeit	BaE	
Prüfungsform und -dauer	Mündliche Präsentation und schriftliche Dokumentation	
Lehr- und Lernmethoden	Seminar	
Modulverantwortlicher	C. Koch	
Qualifikationsziele	Ziel der Vorlesung ist es, Konzepte, Anwendungen und Software-Engineering Aspekte autonomer Systeme (hier: autonome mobile Roboter) zu betrachten.	
Lehrinhalte	Einige der wichtigsten Gebiete zur Realisierung autonomer Systeme sind hierbei Sensorik, Aktorik, Regelungstechnik, Bild- und Signalverarbeitung, Algorithmen- und Datenstrukturen als auch Echtzeitprogrammierung. Aktuelle Anwendungen aus dem Bereich der industriellen und universitären Forschung werden in der Veranstaltung vorgestellt, um aktuelle und zukünftige Einsatzgebiete autonomer Systeme zu veranschaulichen.	
Literatur	tagesaktuell	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
C. Koch	Autonome Systeme	2

Modulbezeichnung	Objektorientierte Methoden zur Hardwaresteuerung	
Dauer	1 Semester	
Art	Wahlpflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)	Programmieren 2	
Empf. Voraussetzungen	Programmieren 3, Java 2	
Verwendbarkeit	BaE	
Prüfungsform und -dauer	Erstellung und Dokumentation von Rechnerprogrammen Mündliche Präsentation und schriftliche Dokumentation	
Lehr- und Lernmethoden	Seminar, Praktikum, Studentische Arbeit	
Modulverantwortlicher	R. Wenzel	
Qualifikationsziele	Die Studierenden sind in der Lage, Möglichkeiten der objektorientierten Programmierung zur Steuerung und Synchronisation von Hardwareansteuerungen einzuschätzen und anzuwenden. Sie können Vergleiche zum prozeduralen Ansatz vornehmen und einen optimalen Entwurf, sowie die zugehörige Implementaion selbständig vornehmen und begründen.	
Lehrinhalte	Es werden grundlegende Prinzipien des Datenaustausches und dabei wesentlicher Aspekte als Einführung wiederholt. Prozesse, die z.B. im Hintergrund laufen (Threads) werden hinsichtlich ihrer Anwendung erläutert und an Beispielen veranschaulicht. Die Studierenden sind in der Lage, Zusammenhänge der Kommunikation und des Programmentwurfs verbinden zu können.	
Literatur	Isernhagen, R: Softwaretechnik in C und C++, Hanser, 2004 Wenzel, R. Vorlesung Programmieren 1 und 2, HS-Emden-Leer, seit 2004	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
R. Wenzel	Objektorientierte Methoden zur Hardwaresteuerung	4

Modulbezeichnung	Schnittstellen und Bussysteme	
Dauer	1 Semester	
Art	Wahlpflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Rechnerarchitektur	
Verwendbarkeit	BaE	
Prüfungsform und -dauer	Klausur 1,5 h Mündliche Präsentation und schriftliche Dokumentation	
Lehr- und Lernmethoden	Vorlesung, Studentische Arbeit	
Modulverantwortlicher	R. Wenzel	
Qualifikationsziele	Die Studierenden sind in der Lage, Schnittstellen und Bussysteme für jeweilige Aufgabenstellungen auszuwählen, zu realisieren und auch zu optimieren. Zusammenhänge einer gewünschten Datenübertragung ermöglichen Aufwands- und Kostenanalysen. Durch den umfassenden Überblick über verschiedenste Möglichkeiten der Kommunikation wird damit eine Entscheidungshilfe gefördert.	
Lehrinhalte	Es werden verschiedene Kommunikationsmöglichkeiten behandelt. Dies erfolgt einerseits über die "traditionellen" Schnittstellen, andererseits über verschiedene Arten von Bussystemen. Hier werden u.a. der IEC-Bus, serielle Bussysteme, eigene Interfaces und die jeweiligen Übertragungsprotokolle veranschaulicht. An simulierten Beispielen werden die Erkenntnisse vertieft.	
Literatur	Dembowski, K.: Computerschnittstellen und Bussysteme, Hüthig, 2001 Schnell, G.: Bussysteme in der Automatisierungs- und Prozeßtechnik, Vieweg, 2000	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
R. Wenzel	Schnittstellen und Bussysteme	4

Modulbezeichnung	Test	
Dauer	1 Semester	
Art	Test 6 Pflichtfach	
ECTS-Punkte	5	
Studentische Arbeitsbelastung	70 h Kontaktzeit + 80 h Selbststudium	
Voraussetzungen (laut BPO)		
Empf. Voraussetzungen	Java 1	
Prüfungsform und -dauer	Klausur 1,5 h	
Lehr- und Lernmethoden	Vorlesung, Praktikum	
Modulverantwortlicher	F. Rump	
Qualifikationsziele	Die Studierenden sollen eine konkrete Problemstellung analysieren und algorithmisch lösen können. Sie kennen wichtige Java-Bibliotheken und können diese für konkrete Aufgabenstellungen anwenden. Die Programme werden auf Basis aktueller Werkzeuge erstellt und getestet. Die Studierenden verstehen das Verfahren der testgetriebenen Entwicklung und können dieses für kleine Beispiele anwenden.	
Lehrinhalte	Auf Basis der in "Java 1" gelegten Grundlagen werden weitergehende Konzepte der objektorientierten Programmierung vorgestellt und die Verwendung objektorientierter Bibliotheken vertieft. Behandelt werden u.a. Datenströme und Dateizugriff, Threads, Netzwerkprogrammierung, Unit-Tests, graphische Benutzungsoberflächen mit vorgegebenen Komponenten und Ereignisverarbeitung. Typische Programmstrukturen werden anhand gängiger Entwurfs- und Architekturmuster (z.B. Model-View-Controller) erläutert.	
Literatur	Ratz, D.; Scheffler, J.; Seese, D.; Wiesenberger, J.: Grundkurs Programmieren in Java, Hanser, 2010. Krüger, G.; Stark, T.: Handbuch der Java-Programmierung, Addison-Wesley, 2009.	
Lehrveranstaltungen		
Dozent	Titel der Lehrveranstaltung	SWS
F. Rump	Java 2	3
F. Rump	Praktikum Java 2	1